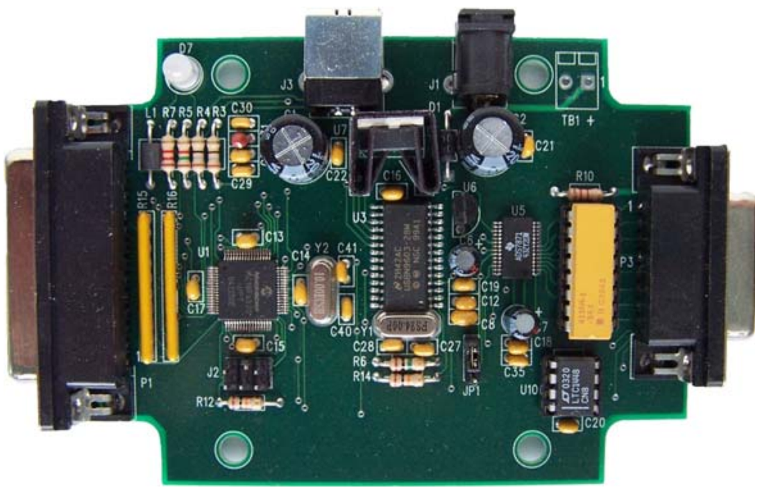
Integrity Instruments

P.O. Box 451
Pine River Minnesota
56474 USA

Order Phone 800-450-2001
Fax Phone 218-587-3414
Tech Phone 218-587-3120

http://www.integrityusa.com



USBM400
User's Manual

Analog and Digital I/O Module

Table of Contents

Introduction

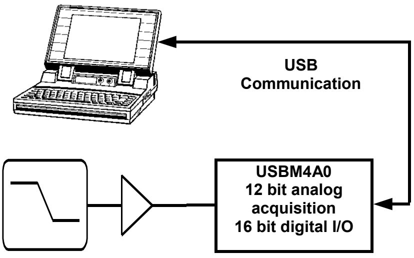
Welcome to the Integrity Instruments **USBM400 Series** of I/O modules. These modules using USB communications are available in different configurations dependent on your needs and applications. In addition they are offered in an enclosure, or open allowing you the end user complete flexibility when determining the parameters for your project.

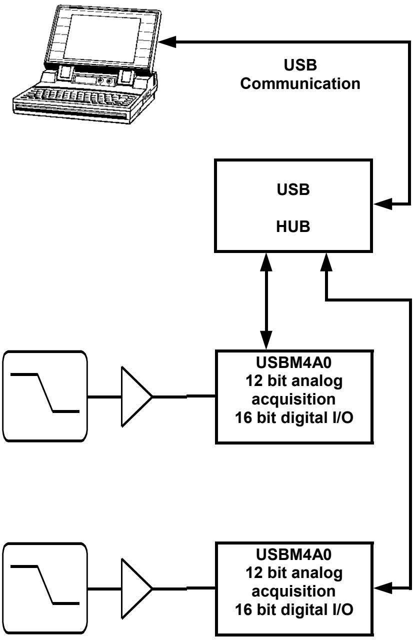
Configurations for USBM400 models with enclosure are:

USBM400CE	16 digital I/O
USBM4A0CE	16 digital I/O and 8 channels A to D conversion
USBM4ADCE	16 digital I/O and 8 channels A to D conversion and 2 channels D to A conversion

I/O Module features:

MPU:	Microchip PIC18F6520
EEPROM:	Flash internal
MPU Clock:	10 Mhz
Interface:	National USBN9603
A to D	Burr Brown ADS7871, 14 bit programmable gain
LED:	Bicolor diagnostic LED
POR:	MPU contains timed Power On Reset circuitry
Brownout:	MPU brownout detection ciruictry built-in
Temperature:	0° to 70°C (32° to 158°F) *Commercial Temperature Range*
PCB:	FR4
Power:	7.5Vdc to 15.0Vdc, approximately 50 ma. (our PS9J)
Cabling	Standard USB **A**-**B** cable.





Windows 2000 Installation

You need the following:

- Integrity Instruments Software CD
- An open USB port on your PC or USB Hub
- Power supply PS9J (9VDC 400 ma unregulated)
- USB cable to connect the USBM400 to your PC or USB Hub

Plug your USBM300 into an available USB port and follow these steps:

Step 1

The USBM400 is plugged in and discovered by the operating system.

Click the “**Next**” button

Step 2

Select “**Search for a suitable driver for my device**”

Insert the Software CD

Click the “**Next**” button

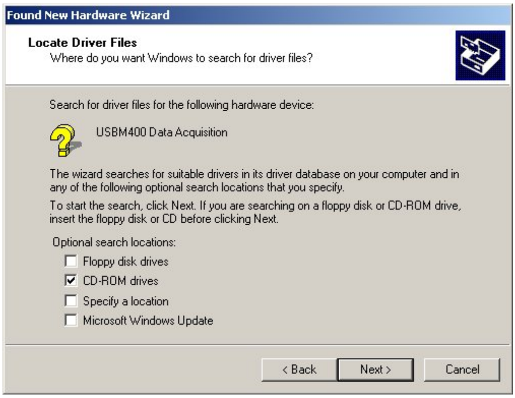
Windows 2000 Installation continued

Step 3

Select “**CD-ROM drives**”

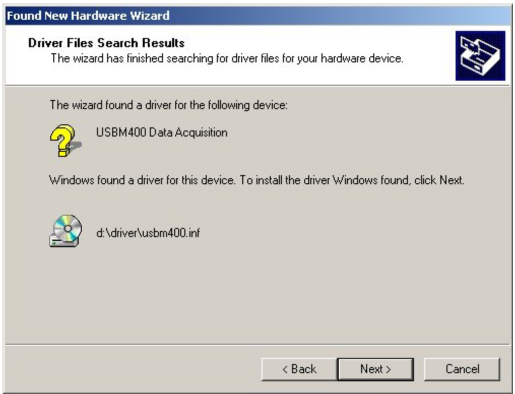
Click the “**Next**” button

Make sure the CD is in the drive.



Step 4

Click the “**Next**” button

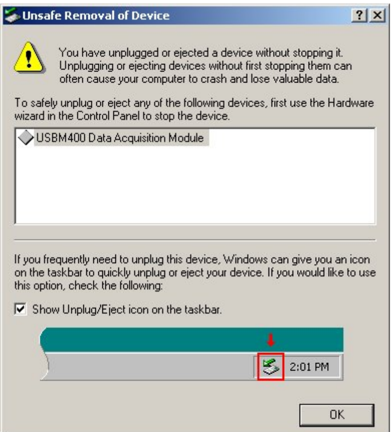


Step 5

Click the “**Finish**” button

The USBM400 driver is now installed and ready to use.

Windows 2000 Device Removal



Unsafe Device Removal Warning

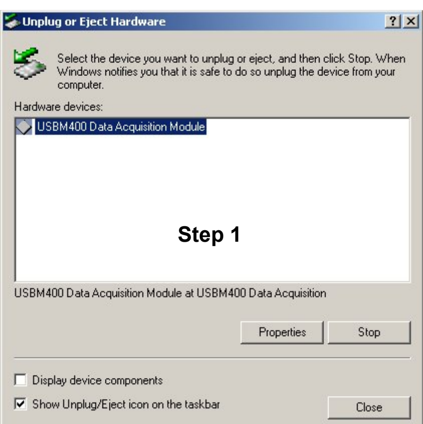
When the USBM400 is unplugged or powered down without first Ejecting the hardware, the Windows 2000 operating system will display a warning to the user.

The USBM400 module should always be stopped before it is unplugged or powered down. This allows Windows to correctly manage the device when it is currently in use.

Once USBM400 has been safely stopped, it can be removed or powered down.

Windows 2000 - Safely Stopping the USBM300 Module

Simply select the "**Safely Remove Hardware**" icon from the Windows 2000 task tray (bottom right hand corner of the desktop display). Highlight the "**USBM400 Data Acquisition Module**" from the list and click on the "**Stop**" button.



Windows XP Installation

You need the following:

- Integrity Instruments Software CD
- An open USB port on your PC or USB Hub
- Power supply PS9J (9VDC 400 ma unregulated)
- USB cable to connect the USBM300 to your PC or USB Hub

Plug your USBM400 into an available USB port and follow these steps:

Step 1

The USBM400 is plugged in and discovered by the operating system.

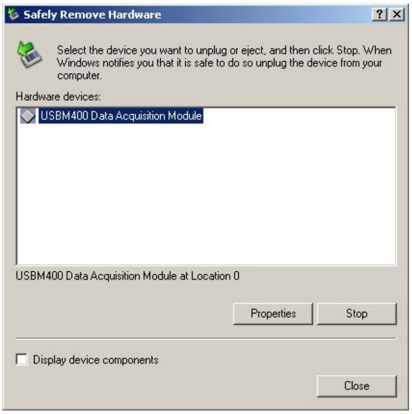
Insert the Software CD.
Click the "**Next**" button

Step 2

Click "**Finish**" button

The USBM400 driver is now installed and ready to use.

Windows XP Device Removal



Windows XP does not display an **Unsafe Device Removal** message when the USBM400 module is disconnected without first stopping the device. Although the warning message is not displayed, **you should always stop the USBM400 device before unplugging.** This allows Windows to correctly manage the device when it is currently in use.

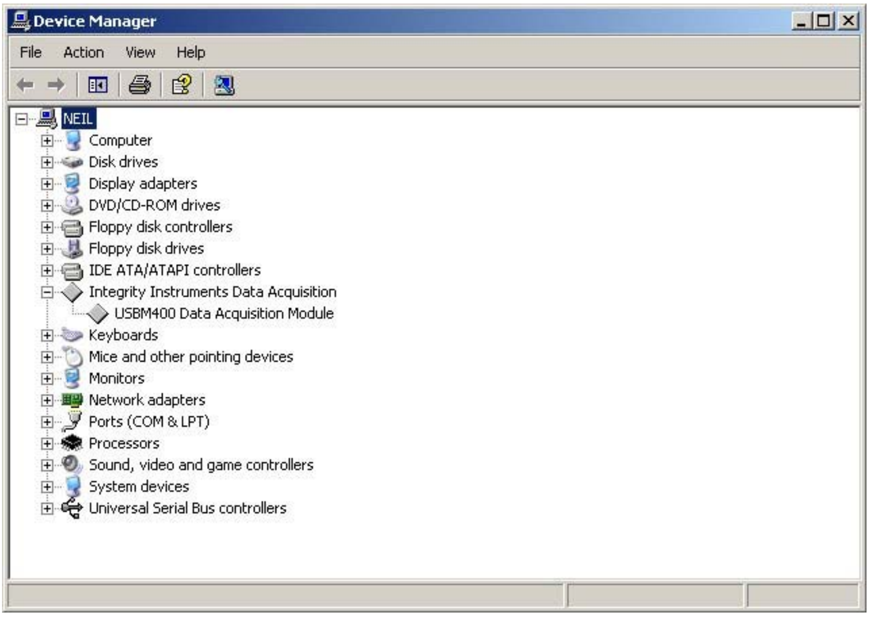
Simply select the "**Safely Remove Hardware**" icon from the Windows XP task tray (bottom right hand corner of the desktop display). Highlight the "**USBM400 Data Acquisition Module**" from the list and click on the "**Stop**" button.

Once USBM400 has been safely stopped, it can be removed or powered down.

Verifying with Device Manager

The USBM400 device installation can be verified with Device Manager. When a "**USBM400 Data Acquisition Module**" entry appears under the "**Integrity Instruments Data Acquisition**", then the device has been installed correctly.

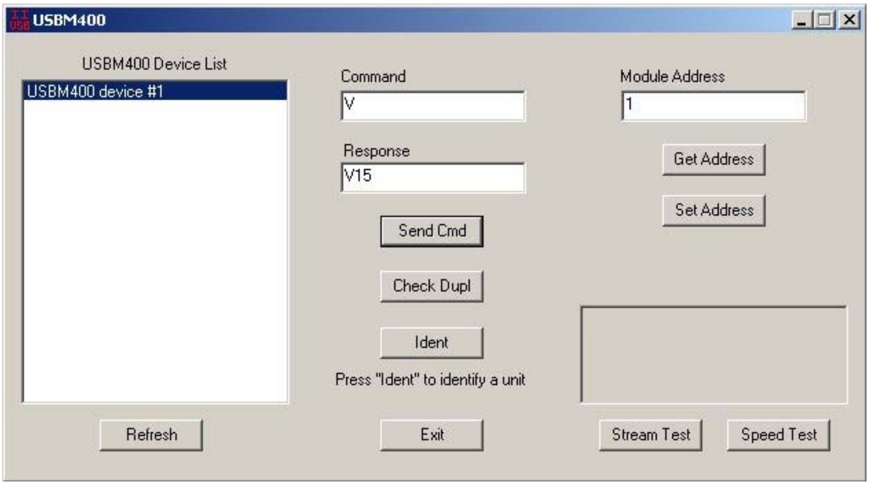
Device Manager can be started by right clicking "My Computer", selecting "Properties" menu, "Hardware" tab and the "Device Manager" button. Or, running "Computer Management" in the administrative tools and selecting "Device Manager".



USBM400 Test GUI (Graphics User Interface)

This program allows the user to quickly and easily send commands to a USBM400 module as well as perform basic utility functions such as setting a module's address, checking for duplicate module addresses and identifying a module where multiple USBM400 modules are used in a system.

Button	Description
Send Cmd	Sends a command and displays the response
Check Dupl	Checks duplicate module addresses. When multiple UISBM300 modules are connected to a system, this allows the user to detect if more than one module is configured with the same module address.
Ident	Start/Stops identifying a USBM400. When multiple USBM400 modules are connected to a system, this allows the user to identify a unit when it is selected in the "**USBM400 Device List**" The module's LED will illuminate RED to show USB activity
Get Address	Obtains the module address
Set Address	Stores a new module address
Refresh	Scans for new USBM400 Device List
Stream Test	Tests the module's Stream mode
Speed Test	Tests the module's Command-Response speed
Exit	Exits the program.



Quick Start

1) Apply power and attach the USBM400 to the host PC. **Be sure the drivers are already installed.** See previous installation instructions.
2) Launch the USBM400_GUI.EXE program.
3) Enter a command in the "**Command**" edit box. Type the letter **V**
4) Select the "**Send Cmd**" button. The response **V16** should appear in the "**Response**" edit box.

You can now send commands to your USBM400 module!

Running USBM400 Tests

The USBM400_GUI program has a few built-in tests to help deploy the USBM400 module in new systems. Performance numbers are obtained by running the Stream Test and the Speed Test. Typically the Speed Test performance is more variable than the Stream Test due to the round-trip behavior of a Command/Response pair.

Test	Description
Stream Test	Uses the USBM400 C++ Class to enter Continuous Stream mode and capture data in the background. The test is executed for 10 seconds and the data received from the USBM400 module is saved in file **STREAM_DATA.TXT** When finished the test displays the total data blocks and bytes received from the USBM400 module.
Speed Test	Uses the USBM400 C++ Class to send commands to the USBM400 module in Command-Response mode. A unipolar analog sample is taken repeatedly on analog CH0 as fast as the machine allows. When finished the test displays the maximum commands per second.

USB Performance Tips

USB 2.0 systems typically perform worse than USB 1.1 systems. The USB 2.0 device driver stacks appear to perform more functionality than the USB 1.1 stacks and as such are more inefficient. Continuous Stream mode performance is not grossly affected by USB 2.0 systems.

1) If Command-Response performance is important, then use USB 1.1 host controllers whenever possible.
2) Reduce the number of bandwidth hungry USB devices like USB audio on the USB bus that the USBM400 module is connected.
3) The best performance is realized by using USB v1.1 compliant host controllers with few additional device that share the same USB bus. USB devices that typically reduce the speed of other USB devices include USB audio and USB storage devices like hard drives.

Commands and Responses v1.4 Firmware

Note: All numeric data is represent as ASCII Hexadecimal integers

Command Sent by Host	**Response** Sent by I/O Module	**Description**
V	Vxy	Firmware version x.y
I	Ixxyy	Input digital port status xx = PORT1 yy = PORT2 ***Also returns current output port status***
Oxxyy	O	Output digital port: xx = PORT1 yy = PORT2
Txxyy	T	Set digital direction: xx = PORT1 yy = PORT2 bit set(1) = Input, bit clear(0) = Output
G	Gxxyy	Get current digital direction: xx = PORT1 yy = PORT2 bit set(1) = Input, bit clear(0) = Output
N	Nxxxxxxxx	Get Pulse Counter (xxxxxxxx 32 bit counter value)
M	M	Clear Pulse Counter
Uzy *See table below*	Uzyxxxx	Unipolar or Bipolar {determined by control nibble} sample analog (z gain y, control nibble, xxxx analog value) **Gain values are in table below**.
Lyxxx	L	D/A output (y channel setting 0 or 1, xxx 12 bit D/A output)
Wyyxx	W	Write EEPROM (yy address, xx value)
Ryy	Rxx	Read EEPROM (yy address in command, xx value in reponse)
S	S	Start continuous stream mode
H	H	Halt continuous stream mode
	X	**Command error response**

GAIN VALUES			
Analog Z value	Gain	Analog Z value	Gain
0	1	4	8
1	2	5	10
2	4	6	16
3	5	7	20

Commands and Responses: Examples
The following table illustrates actual command and response data for a USB interface.

Note:

- All numeric data is represent as ASCII Hexadecimal integers.

Command Sent by Host	**Response** Sent by I/O	**Description**
V	V14	Module Firmware version 1.4
I	IFF00	Input digital port [PORT1 bits0-7 ON] [PORT2 bits0-7 OFF] **Note:** this command also returns the current digital output
O007F	O	Output digital port [PORT1 bits 0-7 OFF] [PORT2 bit 7 OFF, bits 0-6 ON]
TFF80	T	Set digital direction [PORT1 bits 0-7 INPUT] [PORT2 bit 7 INPUT, bits 0-6 OUTPUT]
G	GFF80	Get current digital direction [PORT1 bits 0-7 INPUT][PORT2 bit 7 INPUT, bits 0-6 OUTPUT]
N	N0000000F	Get pulse counter: Current count = 15
M	M	Clear pusle counter: Current count = 0
U01 *See note 1*	**U01**100F	Bipolar analog control nibble = **0x01,** gain of 1 Analog reading = 0x100F
U08 *See note 1*	**U08**140F	Unipolar analog control nibble = **0x08** gain of 1 Analog reading = 0x140F
L1800	L	D/A Output Channel 1 = 2.5 Volts
W0410	W	Write EEPROM Address 0x04 with value 0x10
R04	R10	Read EEPROM Adress 0x04 (value is 0x10)
S	S	**START** continuous stream mode See **“Modes of Operation”** section
H	H	**HALT** continuous stream mode

Note

1. The analog command “U” is followed by 2 digits. The first digit is the gain (see page 11). The second digit is the control nibble (see page 13). The control nibble determines if the sample is differential or single ended.

Analog Control Nibble and Example

The **USBM400 Series** I/O modules equipped with analog inputs utilizes the Burr Brown ADS1781 analog to digital conversion chip. In the process of performing a data sample, the user sends a control nibble to the **USBM400 Series** module. The **USBM400 Series** module in turn performs a data conversion using the control nibble and transmitts a response data sample back. The following table lists each of the 16 possible analog configurations.

Note:

- All numeric data is represent as ASCII Hexadecimal integers
- See **Analog I/O Technical Information** section for sample to volts conversion

Control Nibble	Analog Sample
0	Differential: CH0+ CH1-
1	Differential: CH2+ CH3-
2	Differential: CH4+ CH5-
3	Differential: CH6+ CH7-
4	Differential: CH0- CH1+
5	Differential: CH2- CH3+
6	Differential: CH4- CH5+
7	Differential: CH6- CH7+
8	Single Point: CH0
9	Single Point: CH1
A	Single Point: CH2
B	Single Point: CH3
C	Single Point: CH4
D	Single Point: CH5
E	Single Point: CH6
F	Single Point: CH7

ANALOG EXAMPLE

Command Sent by Host	**Response** Sent by I/O Module	**Description**
U00	**U00**000F	Bipolar sample differential CH0+ CH1- (Gain = **1**, Control = **0**) Analog sample = 0x00F (decimal 15)
U1A	**U1A**0123	Unipolar sample CH2 (Gain = **2**, Control = **A**) Analog sample = 0x123 (decimal 291)

EEPROM Map:

Address	Description
0x00	Module Address **[factory default = 0x01]**
0x01	N/A - Reserved
0x02	Data Direction Port 1 Bit set (1) = Input Bit clear (0) = Output **[factory default = 0xFF]**
0x03	Data Direction Port 2 Bit set (1) = Input Bit clear (0) = Output **[factory default = 0xFF]**
0x04/0x05	N/A - Reserved
0x06	Port 1 Power on Default output **[factory default = 0x00]**
0x07	Port 2 Power on Default output **[factory default = 0x00]**
0x08 *See Note 1*	Expander board flag (Opto-22® modules attached) 0x00 = No expander board attached 0xFF = Expander board attached (invert digital signals) **[factory default = 0x00]**
0x09/0x0A	D/A Channel 0 Power on Default output 12 bits - upper nibble in 0x09, lower byte in 0x0A **[factory default = 0x000]**
0x0B/0x0C	D/A/ Channel 1 Power on Default output 12 bits - upper nibble in 0x0B, lower byte in 0x0C **[factory default = 0x000]**
0x0D *See Note 2*	A/D Channels sample clock rate 0x00 = Normal A/D Channels sample clock rate 0xFF = Slowed A/D Channels sample clock rate **[factory default = 0x00**
0x0E *See Note 3*	ADS7871 Ref voltage 0x0C = 2.5 V 0x0E = 2.048 V 0x0F = 1.15 V 0x04 = External **[factory default]** 0x0C = 2.5V Internal

Notes:

1. This flag is used when an expander board is attached. It allows for polarity interface to the industry standard I/O modules used with the expander board based on open collector logic that these modules use.
2. This is used to slow the A/D Channel sample clock rate. This may help when the A/D channels have a high impedance input attached.
3. There are 4 reference voltages that can be used for the ADS7871 A to D chip. JP1 on the board determines external reference voltage (5.0 V) or internal reference voltage. If you want to use the internal reference voltage JP1 must be positioned to the internal reference position. See page 31 for more information. ***When EEPROM position 0x0E is changed, the unit must have power recycled for this change to take effect.***

EEPROM Map continued:

Address	Description
0x0F0x10	N/A - Reserved
0x11	Analog Sample 1 - control byte - **analog control nibble**
0x12	Analog Sample 2 - control byte **- analog control nibble**
0x13	Analog Sample 3 - control byte **- analog control nibble**
0x14	Analog Sample 4 - control byte **- analog control nibble**
0x15	Analog Sample 5 - control byte **- analog control nibble**
0x16	Analog Sample 6 - control byte **- analog control nibble**
0x17	Analog Sample 7 - control byte **- analog control nibble**
0x18	Analog Sample 8 - control byte **- analog control nibble**
0x19	Analog Sample 9 - control byte - **analog control nibble**
0x1A	Analog Sample 10 - control byte - **analog control nibble**
0X1B	Analog Sample 11 - control byte - **analog control nibble**
0X1C	Analog Sample 12 - control byte - **analog control nibble**
0X1D	Analog Sample 13 - control byte - **analog control nibble**
0X1E	Analog Sample 14 - control byte - **analog control nibble**
OX1F	Analog Sample 15 - control byte - **analog control nibble**
OX20	Analog Sample 16 - control byte - **analog control nibble**
0x21 ... 0xFF	Available to User

EEPROM locations 0x11 to 0x20 are used to configure analog samples for **Continuous Stream** mode data acquisition.

Analog and Digital I/O Sampling Rates

Analog I/O	
Polled Mode	**Stream Mode**
1,000	8,000
Digital I/O	
1,000	N/A

Digital sampling rates are based on digital input of both ports (16 bits) and analog sampling rates in samples per second per channel. Tested on DELL 600SC 2.4 Ghz Pentium 4 with USB v1.1 compliant controller.

Modes of Operation:

The USBM400 I/O modules operate in two communications modes:

1) **Command-Response**
2) **Continuous Stream**

These modes can be used singularly or together in combination.

The USBM400 module uses three different channels or “pipes” of USB communications: two for Command-Response and one for Continuous Stream thereby allowing parallel operation where both Command-Response and Continuous Stream modes run concurrently without affecting one another.

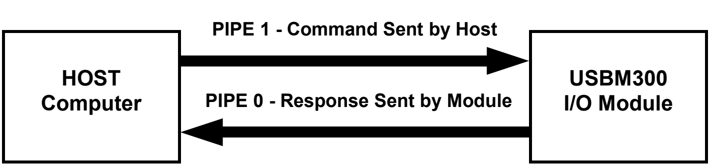
PIPE 1 Command: Host PC ► USBM400
PIPE 0 Response: USBM300 ► Host PC
PIPE 2 Stream: USBM300 ► Host PC

Command-Response Mode

This mode of operation is the simplest and is the first step to use the USBM400 module. The Host computer sends a command to the USBM400 I/O Module which in turn replies with a response back to the Host computer.

All commands are designed to be used in this mode of operation. You will notice that each command in the Command Table has an associated response.

Please see the **Command and Responses** table for more information.



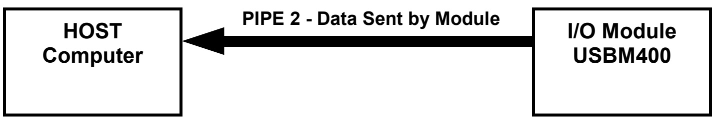
Continuous Stream Mode

The Continuous Stream mode of communication allows the USBM400 module to constantly send or ***stream*** data to the host. In brief, the I/O Module sends data blocks that consist of 32 analog samples.

The HOST may send any command (in Command-Response mode) while Continuous Stream mode is in operation and it will be accepted and processed by the I/O Module as in normal operation.

The I/O module uses parameters found in EEPROM locations 0x11 thru 0x20 to configure and control the continuous stream mode analog sampling.

The EEPROM must be configured before engaging the Continuous Stream mode.



Continuous Stream Mode Data Block

The Continuous Stream mode data is sent by the USBM400 I/O module in blocks. The blocks are defined as:

1) 16 A/D samples are taken 2 times back-to-back for a total of 32 samples
2) Digital input status (both 8 bit ports). See command **Ixxyy**
3) Pulse counter status (32 bit value). See command **Nxxxxxxxx**

```
/////////////////////////////////////////////////////////////
// Stream mode data block sent by USBM300
// Each block consists of:
// 24 12 bit A/D samples (stored as 16 bits each)
// 1  16 bit Digital Input Status
// 1  32 bit Pulse Counter Status
// 54 total bytes
/////////////////////////////////////////////////////////////
typedef struct _USBM300_STREAM_BLOCK
{
    USHORT  usAdcSamples[ADC_SAMPLE_COUNT];
    USHORT  usDigitalInput;
    ULONG   ulPulseCounter;
} USBM300_STREAM_BLOCK, *PUSBM300_STREAM_BLOCK;
```

Continuous Stream mode setup steps

1. Configure EEPROM locations 0x11 thru 0x20
2. Either call the USBM400 C++ Class to start streaming by calling the **ControlStream-ing()** member function or the USBM400 Active X control.

Continuous Stream Mode Configuration Example— EEPROM Locations

All parameters configuring the Stream Continuous mode are strored in EEPROM addresses 0x11 to 0x20. See the following table for a description of the locations and the parameters. Use command **'W'** to update EEPROM values.

These values represent the analog control nibble for 16 analog samples that are taken 2 times back-to-back. See **Analog Control Nibble and Example** section earlier in this manual for a discussion on the analog control nibble.

Sample Configuration Bits *y*	Description
Bits 0 to 3	Analog control nibble
Bits 4 to 6	Gain
Bit 7	Unused

EEPROM Location	Value	Description
0x11 *Sample 1*	0x00	Gain = 1 Bipolar Analog: CH0+ CH1-
0x12 *Sample 2*	0x10	Gain = 2 Bipolar Analog: CH0+ CH1-
0x13 *Sample 3*	0x08	Gain = 1 Unipolar Analog: CH0
0x14 *Sample 4*	0x18	Gain = 2 Unipolar Analog: CH0
0x15 *Sample 5*	0x2A	Gain = 4 Unipolar Analog: CH2
0x16 *Sample 6*	0x0B	Gain = 1 Unipolar Analog: CH3
0x17 *Sample 7*	0x0C	Gain = 1 Unipolar Analog: CH4
0x18 *Sample 8*	0x0D	Gain = 1 Unipolar Analog: CH5
0x18 *Sample 9*	0x0E	Gain = 1 Unipolar Analog: CH6
0x18 *Sample 10*	0x0F	Gain = 1 Unipolar Analog: CH7
0x18 *Sample 11*	0x05	Gain = 1 Bipolar Analog: CH2- CH3+
0x18 *Sample 12*	0x26	Gain = 4 Bipolar Analog: CH4- CH5+
0x18 *Sample 13*	0x48	Gain = 8 Unipolar Analog: CH0
0x18 *Sample 14*	0x01	Gain = 1 Bipolar Analog: CH2+ CH3-
0x18 *Sample 15*	0x39	Gain = 5 Unipolar Analog: CH1
0x18 *Sample 16*	0x59	Gain = 10 Unipolar Analog: CH1

Continuous Stream Mode Configuration — Using Module Setup

The easiest way to configure the USBM400 module for Continuous Stream Mode is by using the Module Setup program to configure the analog samples.

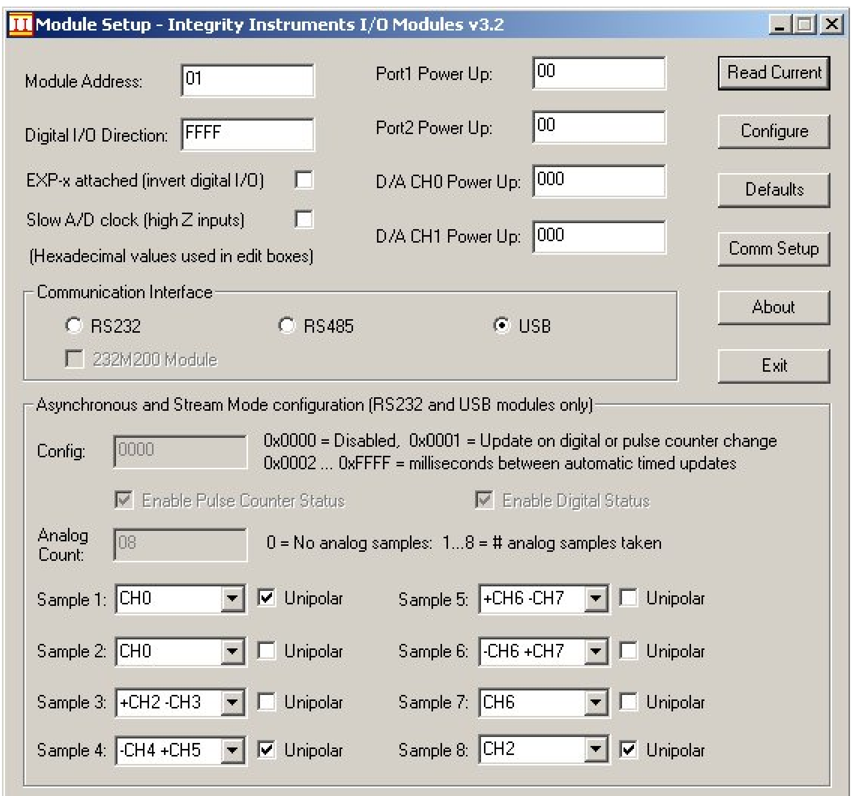
The USBM400 module always takes 16 analog samples 2 times back-to-back for a total of 32 analog samples. The samples can be configured in whatever fashion the user requires by using Module Setup to select the analog channel and sample type (Bipolar/ Unipolar).

Selecting Analog Channel

The analog channel is selected by using a list drop box. The user can select any channel or mix of channels for all 8 analog samples including both single-ended (CH0 ... CH7) or differential (+CHx -CHy or -CHx +CHy).

Unipolar/Bipolar A/D Samples

When the **Unipolar** box is checked, then the analog sample will be performed Unipolar (0 ... 5V). If the **Unipolar** box is **not** checked, then the analog sample will be performed Bipolar (-5V ... 5V)



Digital I/O Characteristics

The following chart lists the Digital I/O characteristics and values.

Characteristic	Value
Digital I/O Current	I/O line source & sink 25 ma Total current PORT1 200 ma Total current PORT2 200 ma
Digital I/O Voltage Levels	Input Off (0) = 0V - 0.8V Input On (1) = 2.0V - 5.0V Output Off (0) = 0.6V max. Output On (1) = 4.3V min.
Pulse Counter Input	1 Mhz max. input rate 32 bit counter capture **Counter increments on high-low transition**

Digital Port Configuration Example

Any Digital I/O configuration changes made to the I/O Module using the '**T**' command are stored in EEPROM locations 0x02 and 0x03.

EEPROM Location 0x02	Port 1 I/O Configuration
EEPROM Location 0x03	Port 2 I/O Configuration

When using either the '**T**' command or directly writing to EEPROM using the '**W**' command, a binary 1 at a bit location puts the I/O line into Input mode, while a binary 0 at a bit location puts the I/O line into Output mode.

Note:

- All numeric data is represent as ASCII Hexadecimal integers

Host Command	Module Response	Action
T0000	T	All I/O lines are configured as Outputs
TFFFF	T	All I/O lines are configured as Inputs
TFF00	T	Port 1 bits 0-7 Inputs Port 2 bits 0-7 Outputs
T00FF	T	Port 1 bits 0-7 Outputs Port 2 bits 0-7 inputs
T1234	T	Port 1 bits 4,1 Inputs Port 1 bits 7,6,5,3,2,0 Outputs Port 2 bits 4,5,2 Inputs Port 2 bits 7,6,3,1,0 Outputs

Hexadecimal Conversion Chart

The following hexadecimal conversion chart will help you program the Digital I/O ports.

Commands such as Digital Input (Ixxyy), Digital Output (Oxxyy) and Digital Direction (Txxyy) all use a Hexadecimal representation of the digital port bits for Port 1 and Port 2.

Example:

OC8B7 Digital Output Command (Oxxyy)
Port 1 bits 7,6,3 set On, bits 5,4,2,1,0 set Off
Port 2 bits 7,5,4,2,1,0 set On, bits 6,3 set Off

	Example Hexadecimal Conversion															
	X				X				Y				Y			
Binary 1 = On 0 = Off	1	1	0	0	1	0	0	0	1	0	1	1	0	1	1	1
Hexadecimal	C				8				B				7			

PORT 1										PORT 2									
X					X					Y					Y				
Hexadecimal	Binary Bit Value 1 = ON 0 = OFF				Hexadecimal	Binary Bit Value 1 = ON 0 = OFF				Hexadecimal	Binary Bit Value 1 = ON 0 = OFF				Hexadecimal	Binary Bit Value 1 = ON 0 = OFF			
	7	6	5	4		3	2	1	0		7	6	5	4		3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	0	1
2	0	0	1	0	2	0	0	1	0	2	0	0	1	0	2	0	0	1	0
3	0	0	1	1	3	0	0	1	1	3	0	0	1	1	3	0	0	1	1
4	0	1	0	0	4	0	1	0	0	4	0	1	0	0	4	0	1	0	0
5	0	1	0	1	5	0	1	0	1	5	0	1	0	1	5	0	1	0	1
6	0	1	1	0	6	0	1	1	0	6	0	1	1	0	6	0	1	1	0
7	0	1	1	1	7	0	1	1	1	7	0	1	1	1	7	0	1	1	1
8	1	0	0	0	8	1	0	0	0	8	1	0	0	0	8	1	0	0	0
9	1	0	0	1	9	1	0	0	1	9	1	0	0	1	9	1	0	0	1
A	1	0	1	0	A	1	0	1	0	A	1	0	1	0	A	1	0	1	0
B	1	0	1	1	B	1	0	1	1	B	1	0	1	1	B	1	0	1	1
C	1	1	0	0	C	1	1	0	0	C	1	1	0	0	C	1	1	0	0
D	1	1	0	1	D	1	1	0	1	D	1	1	0	1	D	1	1	0	1
E	1	1	1	0	E	1	1	1	0	E	1	1	1	0	E	1	1	1	0
F	1	1	1	1	F	1	1	1	1	F	1	1	1	1	F	1	1	1	1

Hexadecimal Conversion Chart

Analog I/O Characteristics:

Characteristic	Value
A/D Converter	Burr Brown ADS7871
Integral Linearity Error	± 2 LSB
Temperature Drift	10 PPM/°C Typical
Differential Linearity Error	± 0.05 LSB
Full Scale Gain Error	± 0.02%
Offset Error	± 2 LSB @ REF = 2.5 VOLT
Resolution	14 LSB
Max Input Voltage/Current	5 Volt 10 Milliamps
D/A Converter	Linear Tech LTC1448
Offset Error	± 10 mv

ADS7871 Operation

The A to D converter is a capacitor sample and hold type. Large external source resistors and capacitances will slow the settling of the inputs. It is important that the overall RC time constant is short enough to allow the analog inputs to settle completely within the allowed time.

Sampling Analog Voltage Inputs

By far the most common configuration of the **USBM400 Series** I/O modules is to sample voltage values. Analog voltage levels are converted to integer digital values using the Burr Brown ADS7871 A/D (Analog/Digital) chip.

The input voltage range is determined by the reference voltage.

Vref = 5.000 standard (2.5 volt, 2.048 volt and 1.15 volt available *see page 14*)

A/D sampling is a 14 bit binary integer value, which is in **twos' complement format.**

The high order bit is the polarity or sign bit, so there are 13 bits for the analog data sample value. 13^2 = 8192 For calculations 8191 is used.

Reference Voltage

Dependent on the analog signal amplitude you want to sample, as can be seen from the table below, the lower Vref value, the more resolution to your sample. This would be one of the deciding factors when you pick your Vref level.

1 LSB unipolar = Vref/8191			
Vref = 5.0	Vref =2. 5.0	Vref =2.0485.0	Vref = 1.15.0
5/8191	2.5/8191	2.048/8191	1.15/8191
0.000610426	0.000305213	0.000250031	0.000140398

Analog Sampling

There are two analog sample types:

1) **Unipolar**
2) **Bipolar**

Unipolar Analog Sampling Resolution

Unipolar analog sampling span is from ground (GND) to voltage reference (Vref). **Only positive voltages are sampled in unipolar mode.** The unipolar sample is represented as an unsigned integer as follows:

Unipolar voltages: 0V ... +Vref

The benefit of using Unipolar samples over Bipolar samples is that a 14 bit binary value is spread out over less total voltage span (Vref total.)

1 LSB bipolar = Vref/8191
1 LSB bipolar = 5.000/8191
1 LSB bipolar = 0.000610426 volt

Bipolar Analog Sampling Resolution

Bipolar analog sampling span is from -Vref to +Vref. Both negative and positive voltages are sampled and represented as a signed binary integer (2's complement) as follows:

Bipolar voltages: -Vref ... 0 ... +Vref

The benefit of using Bipolar sampling over Unipolar is obvious, negative voltages! The downfall of using Bipolar sampling is that a 14 bit binary value is spread out over a larger total voltage span (2*Vref total.)

1 LSB bipolar = Vref/8191
1 LSB bipolar = 5.000/8191
1 LSB bipolar = 0.000610426 volt

Voltage Conversion

The Analog conversion value obtained from the **USBM400 Series** module is represented as an integer value (either signed for Bipolar samples or unsigned for Unipolar sample) and is normally converted to a Real or Floating Point number for ultimate usage.

Vref = 5.0 or = 2.5 or = 2.048 or = 1.15

Unipolar Voltage Conversion Formula

Volts [unipolar] = ADC_Sample * (5.0/8191) or (2.5/8191) or (2.048/8191) or (1.15/8191) dependent on which Vref you select.

... or with the voltage per count pre-calculated ...

Volts [unipolar] = ADC_Sample * (0.000610426) or (0.000305213) or (0.000250031) or (0.000140398) dependent on which Vref you select.

Bipolar Voltage Conversion Formula

The following assumes that ADC_Sample is an unsigned integer value.

```
if (ADC_Sample >= 8192)
        Volts [bipolar] = (ADC_Sample-16383) * (5.000/8191)

if (ADC_Sample <= 8191)
        Volts [bipolar] = ADC_Sample * (5.000/8191)
```

... or with the voltage per count pre-calculated ...

```
if (ADC_Sample >= 8192)
        Volts [bipolar] = (ADC_Sample-16383) * 0.000610426

if (ADC_Sample <= 8191)
        Volts [bipolar] = ADC_Sample * 0.000610426
```

Analog Output (D/A) Voltage Formula

Volts out = D/A Setting * (5.000/4096)

Example: command L1800 (D/A output channel 1)

Volts out = 0x800 * (5.000/4096)
Volts out = 2048 * (5.000/4096)
Volts out = 2.500 V

Sampling Current (4-20 ma) Inputs
Many devices output a current value instead of a voltage value. The secret to obtaining current readings is a 250 ohm resistor. Placing a 250 ohm resistor to ground on a 4-20 ma. current input will create a voltage potential of 1V to 5V.

If we remember Ohm's law: E = I * R

R = 250 ohms
I = .004 to .020 amps (4-20 ma.)
E = 1.0V to 5.0V

Obtaining current readings is a three step process:
1. Perform analog Unipolar sample
2. Convert unipolar sample to volts
3. Convert voltage to amps

The following formula will convert the raw analog sample reading to a current value.

Current = (ADC_Sample * (5.000/8192)) / 250

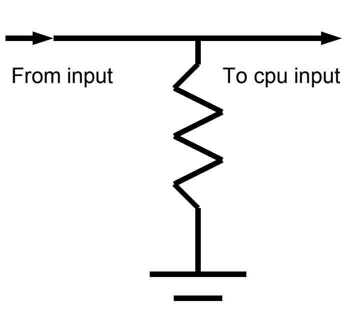
Obtaining accurate Analog samples
Please keep the following points in mind when attempting to obtain accurate samples.

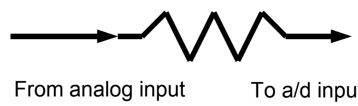
- Avoid high impedance analog signal sources!
- Watch out for UPS systems! They create loads of EMI/EMF noise when operating.
- Keep the analog signal source as close to the USBM400 module as possible.
- Keep transformers far away from the **USBM400** module.
- Use good wiring practices, especially in regards to ground connections.

Resistors for Analog and Digital I/O

The digital I/O points have a 100K Ω resistor to ground to prevent floating inputs.

The analog inputs have a 22 Ω resistor in series to afford some protection to the A to D converter.



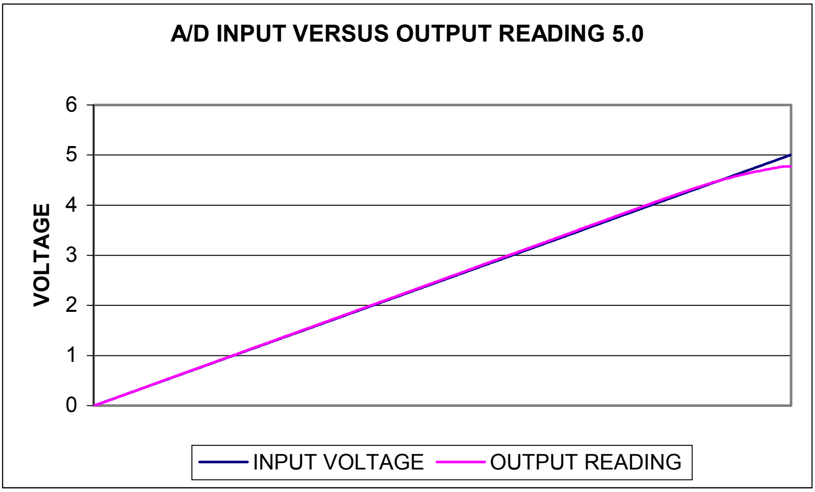


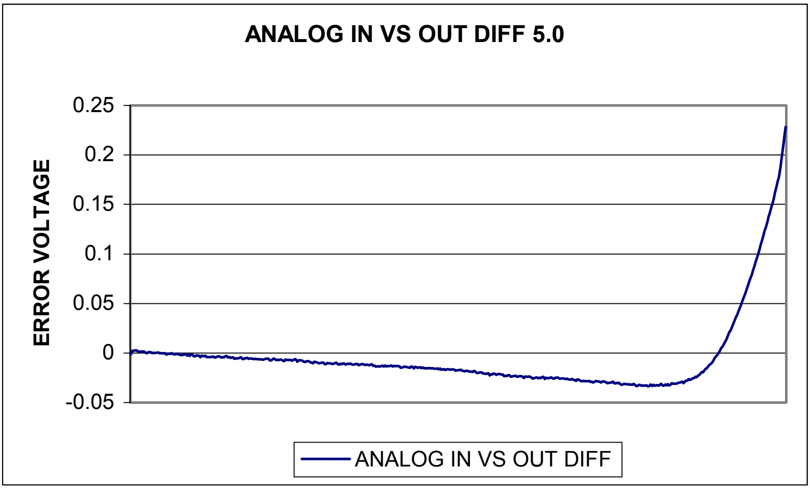
LINEARITY 5.0 Vref

The graphs below show the typical curves for the USBM400 unit with a Vref of 5.0 volts. The first graph is input values plotted against output samples. Note the non linearity when the input voltage nears about 4.5 volt.

The second graph shows the difference between the input voltage, and the output sample. Note the knee that is happening around 4.5 volts.

Keep in mind if your analog inputs are such, that you can use one of the other available voltage references, the readings will be more accurate, as the resolution is greater.



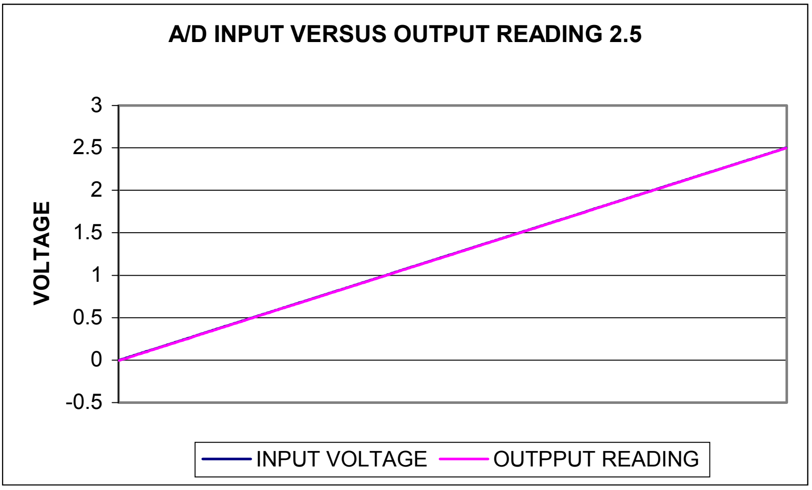


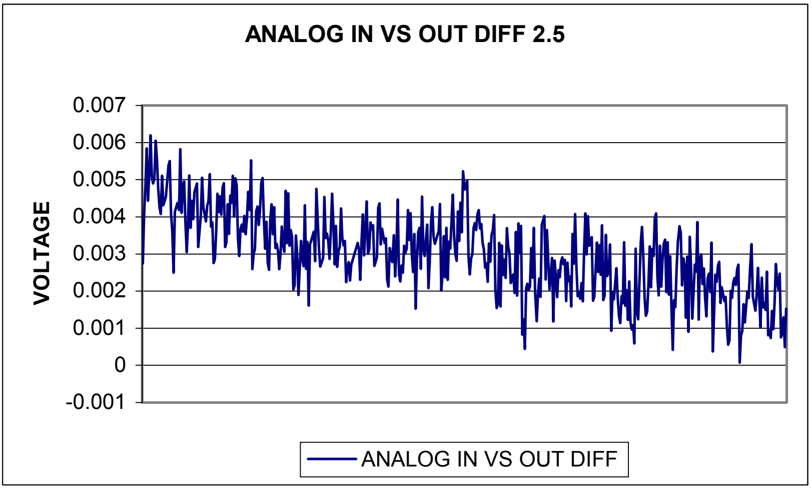
LINEARITY 2.5 Vref

The graphs below show the typical curves for the USBM400 unit with a Vref of 2.5 volts. This is the factory default. The first graph is input values plotted against output samples.

The second graph shows the difference between the input voltage, and the output sample.

Keep in mind if your analog inputs are such, that you can use one of the other available voltage references.



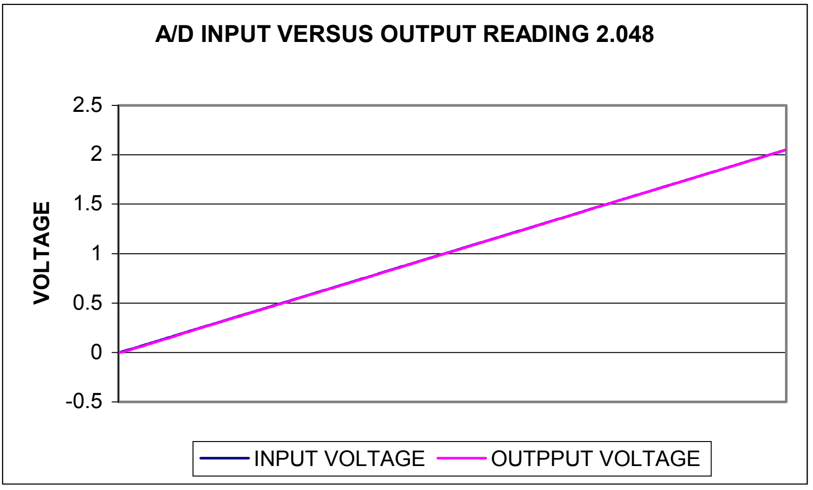


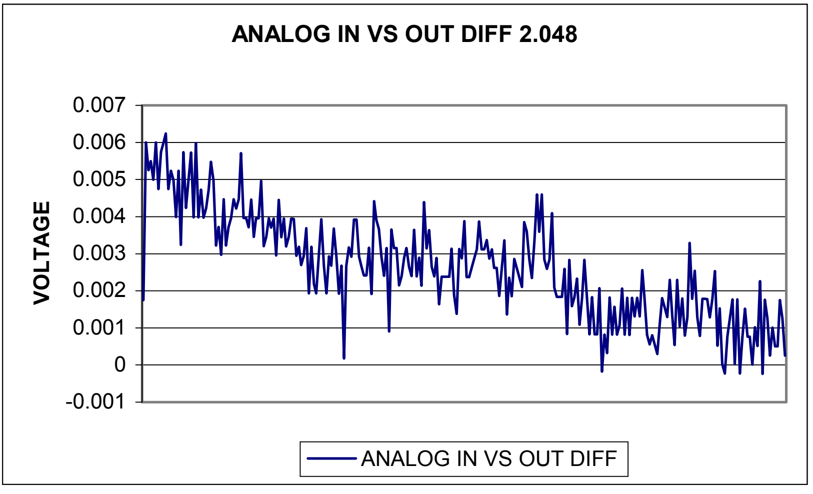
LINEARITY 2.048 Vref

The graphs below show the typical curves for the USBM400 unit with a Vref of 2.048 volts. The first graph is input values plotted against output samples. This Vref gives you an even ratio to sampling. 8192 / 2048 = 4.

The second graph shows the difference between the input voltage, and the output sample.

Keep in mind if your analog inputs are such, that you can use one of the other available voltage references.



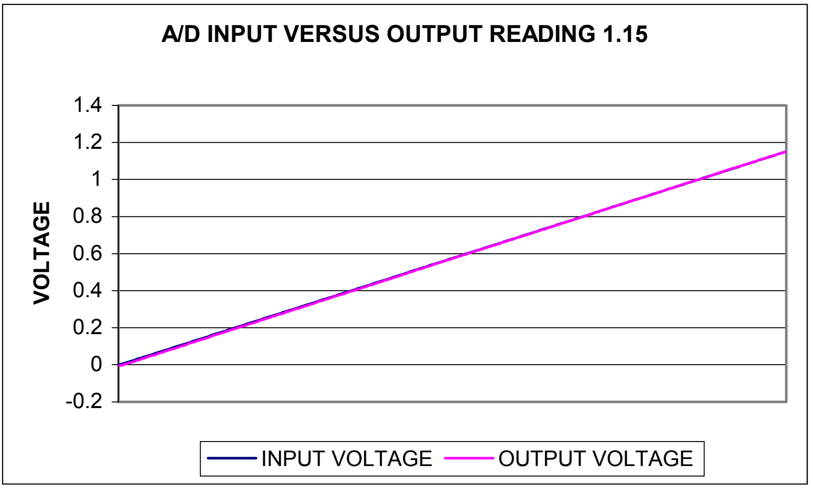


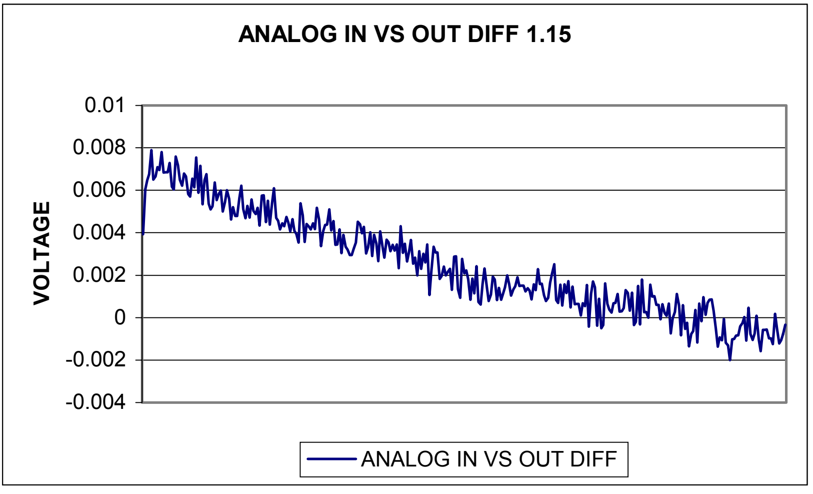
LINEARITY 1.15 Vref

The graphs below show the typical curves for the USBM400 unit with a Vref of 1.15 volts. The first graph is input values plotted against output samples.

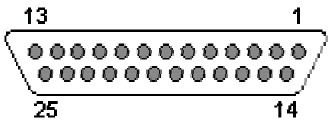
The second graph shows the difference between the input voltage, and the output sample.

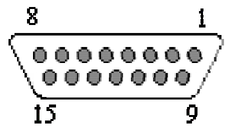
Keep in mind if your analog inputs are such, that you can use one of the other available voltage references.





Digital & Analog I/O Port Pinout

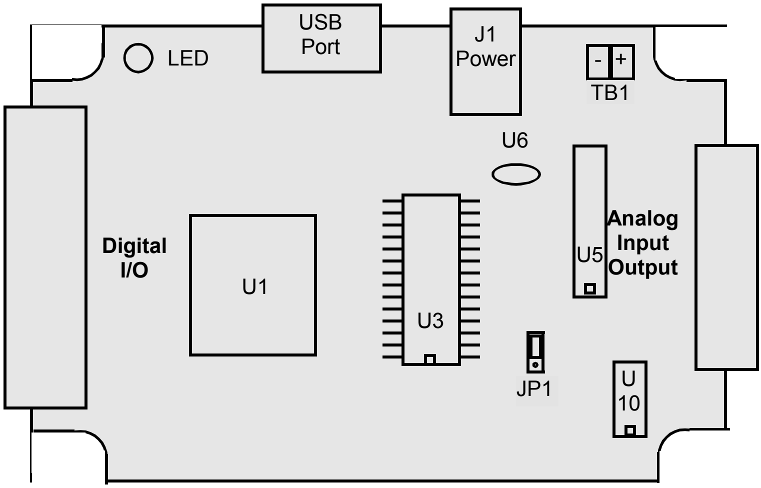




Digital I/O DB25F	
DB25 Pins	**Description**
1	Port 2 Bit 0
2	Port 2 Bit 1
3	Port 2 Bit 2
4	Port 2 Bit 3
5	Port 2 Bit 4
6	Port 2 Bit 5
7	Port 2 Bit 6
8	Port 2 Bit 7
9	PWM output
10	N/A
11	+V Unreg
12	+5Vdc
13	GND
14	Port 1 Bit 0
15	Port 1 Bit 1
16	Port 1 Bit 2
17	Port 1 Bit 3
18	Port 1 Bit 4
19	Port 1 Bit 5
20	Port 1 Bit 6
21	Port 1 Bit 7
22	Pulse Counter Input
23	N/A
24	+5Vdc
25	GND

Analog I/O DB15F	
DB15 Pins	**Description**
1	ANALOG IN CHANNEL 7
2	ANALOG IN CHANNEL 6
3	ANALOG IN CHANNEL 5
4	ANALOG IN CHANNEL 4
5	ANALOG IN CHANNEL 3
6	ANALOG IN CHANNEL 2
7	ANALOG IN CHANNEL 1
8	ANALOG IN CHANNEL 0
9	GND
10	+ V UNREG
11	+ 5VDC REG
12	N/A
13	V REFERENCE
14	ANALOG OUT B
15	ANALOG OUT A

Model USBM400 Series



JP1 position for external reference voltage

JP1 position for internal reference voltage **[factory default]**

Note

When using the ADS7871 external reference voltage, the jumper JP1 must be moved to the external reference position. The internal voltage reference desired must be put into EEPROM position 0x0E. See page 14 for more information.

USBM Series Module Specification

IC	Model **USBM4AD** (Position and type is the same for all models)
U1	PIC18F6520 MPU
U3	USBN9603 USB [28 pin Surface Mount]
U5	ADS7871 A to D [28 pin Surface Mount]
U6	LM4040AIZ-5.0 0.1% Voltage Reference
U10	LTC1448 D to A [8 pin DIP]

LED Operation

Blinking Green	[1 per Second]	Unit functioning correctly. Idle.
Blinking Red	[Rapid]	USB communication.
Solid Green	[no blink]	USBM400 connected to un-powered USB bus or PC.
Solid Red	[no blink]	USBM300 USB error.
No LED		Unit is not functioning.

Peripheral Add-On Modules

AE-8CH	8 channel analog connection board
ASC-2CH	2 channel signal conditioner
DB15TSM	DB15 terminal strip (for analog connector)
DB25TSM	DB25 terminal strip (for digital connector)
EXP-x	Digital Interface board

Model: AE-8CH Analog Connection Board

Jumper configurable analog inputs:

1) 4-20 ma inputs
2) +/- 10 Vdc inputs
3) Solid state temperature probes

Handy terminal strip for all analog connections and voltages. MTA .100 jacks are also available for solid state temperature probes available from Integrity Instruments.

Model: ASC-2CH Signal Conditioning Board

2 channels of precision instrumentation amplifiers.
Gains of 1, 10, 100, 1000

Handy terminal strip for all analog connections and voltages.

Models: DB15TSM and DB25TSM DB Terminal Strip

Terminal strip boards to conveniently connect to DB15 and DB25 connectors.

Models: EXP-x Digital Interface Board

The **EXP-X** unit provides for digital interface and signal conditioning via industry standard opto-isolated I/O modules such as Opto-22. Each unit has 4 I/O points with large easy to use terminal screws. If more I/O points are required, simply plug in another unit up to 16 total I/O points. **Opto isolated modules:** 90V-140V AC input, 12V-140V AC output, 3.3V-32V DC input, 3V-60V DC output.

WARRANTY

Integrity Instruments warranties **all** products against defective workmanship and components for the life of the unit. Integrity Instruments agrees to repair or replace, at it's sole discretion, a defective product if returned to Integrity Instruments with proof of purchase. Products that have been mis-used, improperly applied, or subject to adverse operating conditions fall beyond the realm of defective workmanship and are not convered by this warranty.

Revision: February 7, 2005 - v3.0