



Multifunction Data Acquisition USB Module USB2AD



Multifunction USB Module

USB2AD

AREPOC s.r.o. company would like to thank you for purchasing the Multifunction USB module USB2AD. The information and instructions contained in this manual describes how to install, set up and operate this high quality measurement instrument. We believe it will meet your expectations.

In case you need any assistance, please do not hesitate to contact us:

Address:

AREPOC s.r.o.
ILJUSINOVA 4
851 01 BRATISLAVA
SLOVAKIA

Phone/Fax: +421 2 6382 4613

E-mail: arepoc@zmail.sk

Product No.:

USB chip ID:

The Multifunction USB module USB2AD is specially designed for applications that need a programmable DC current source and precise differential analog inputs. The board in combination with our UsbDAQ software and AREPOC's Hall probes is ideal suited for high-resolution magnetic field measurements.

Major component of this multifunction USB module consists of a single-chip microcomputer, which controls key functions of the analogue and digital inputs/outputs. The module consists of an isolated high resolution integrating AD converter with 8 differential inputs, programmable DC current source, 2 digital inputs and 4 digital outputs.

Conception of this high performance USB module approaches accuracy of 6-1/2 digit 8 channels digital multimeter.

Following are the system requirements for installing the USB module and accompanying software:

- Pentium class PC with a free USB (Ver.2.0) port
- Windows[®] 2000/XP/7 32bit or 64bit operating system

1.1 Unpacking

- The package includes the following:
- Multifunction USB Module USB2AD
- External 9V DC power supply (optional)
- Technical Products CD-ROM with drivers and manuals
- Printed version of the USB2AD manual
- CANNON 25-pin male connector with 10 Ω resistor connected to the current source (pins **14**, **1**) and input Channel 0 (pins **5**, **17**).

Remove the USB module from the package and inspect it for loose components or any other sign of damage. Notify AREPOC if the module appears damaged in any way. Do not connect a damaged module into USB port of your computer.

MULTIFUNCTION USB MODULE USB2AD

Electrostatic discharge can damage several components of the USB module. Never touch the exposed pins of connectors or USB port to avoid damage of the module! !

Typical power output in a PC is 500 mA for mastered USB port, and 100 mA for slaved port, respectively. The USB2AD module (with 100 mA current source) requires totally 400 mA. This means, that if the module is connected to the 100 mA USB port (or USB hub), it will need an external power supply. Multiple USB2AD modules can be connected to same desktop computer or notebook.



Fig. 1.1 Front panel layout



Fig. 1.2 Rear panel layout

2.0 Installing USB2AD Drivers under Windows XP

- If you are running Windows XP or Windows XP SP1, temporarily disconnect your PC from the Internet. This can be done by either removing the network cable from your PC or by disabling your network card by going to the "Control Panel\Network and Dial-Up Connections", right-clicking on the appropriate connection and selecting "Disable" from the menu. The connection can be re-enabled after the installation is complete. This is not necessary under Windows XP SP2/SP3 if configured to ask before connecting to Windows Update. Windows XP SP2/SP3 can have the settings for Windows Update changed through "Control Panel\System" then select the "Hardware" tab and click "Windows Update".
- Connect the device to a spare USB port on your PC. Windows Found New Hardware Wizard will launch. If Windows XP is configured to ask before connecting to Windows Update, the screen shown in *Figure 2.1* is displayed. Select "**No, not this time**" from the options available and then click "**Next**" to proceed with the installation.



Figure 2.1

- Select "**Install from a list or specific location (Advanced)**" as shown in *Figure 2.2* below and then click "**Next**".



Figure 2.2

- Select "**Search for the best driver in these locations**" and enter your CD-ROM path in the combo-box (e.g. E:\ in *Figure 2.3* below) or browse to it by clicking the **Browse** button. Once the file path has been entered in the box, click **Next** to proceed.

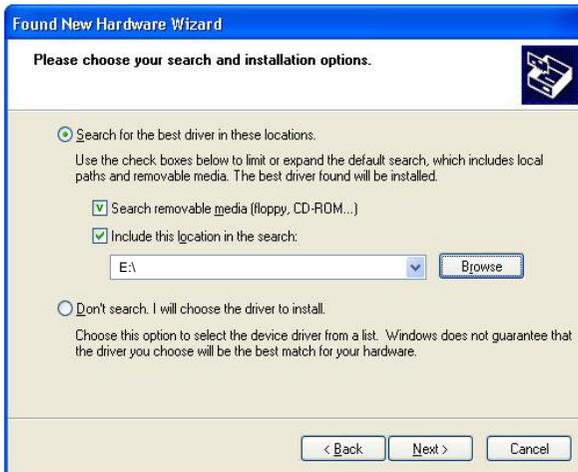


Figure 2.3

- If Windows XP is configured to warn when unsigned (non-WHQL certified) drivers are about to be installed, the message dialogue shown in *Figure 2.4* will be displayed unless installing a Microsoft WHQL certified driver. Click on "**Continue Anyway**" to continue with the installation.



Figure 2.4

- Windows should then display a message indicating that the installation was successful (*Figure 2.5*). Click "**Finish**" to complete the installation for the first port of the device.



Figure 2.5

- Now, the Found New Hardware Wizard will continue by installing the COM port emulation driver for the USB2AD device. The procedure for installing the COM port is identical to that for installing the USB Serial Converter from the first screen of the Found New Hardware Wizard. After repeating the installation procedure, Windows should finally display a message indicating that the installation of USB Serial Port was successful (*Figure 2.6*). Click "**Finish**" to complete the installation for the second port of the device.



Figure 2.6

- Open the Device Manager (located in "Control Panel\System" then select the "Hardware" tab and click "Device Manger") and select "View > Devices by Connection", the device appears as a "USB Serial Converter" with an additional COM port with the label "USB Serial Port" (*Figure 2.7*).

MULTIFUNCTION USB MODULE USB2AD

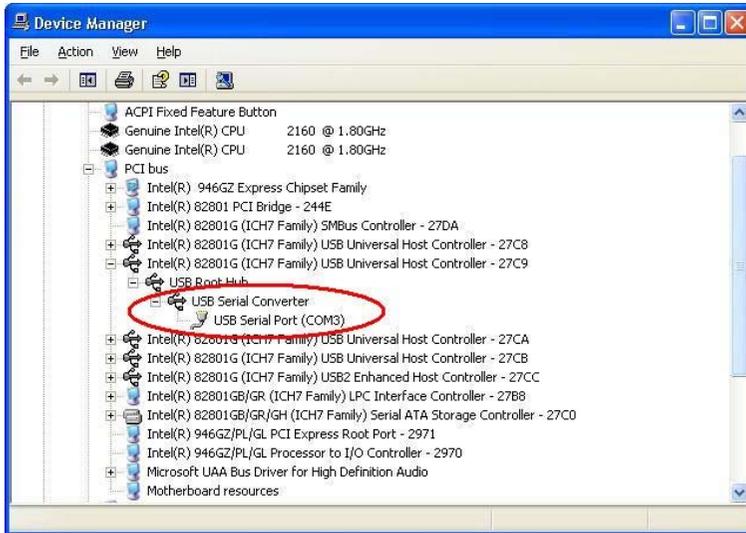


Figure 2.7

Now the first part of your installation has been completed and ready to install supporting software for the module or write your own application.

For the supporting software installation (e.g. UsbDAQ software) follow instruction given by the software installation wizard.

For customers who are writing their own applications, we provide function 32-bit *ad24usb.dll* library compatible for Windows XP/7 operating systems. The *ad24usb.dll* library must be copied to a folder where an application will be located. The DLLs is stored on the *Arepec Technical Products* CD-ROM supplied with the module in the <CD>:\INF directory.

The developing environment can be Visual Basic, Visual C++, Delphi, C++ Builder, LabVIEW, or any Windows 32-bit programming language that allows calls to a DLL. The function reference description of the *ad24usb.dll* is located on the page 22 of this manual.

2.1 Troubleshooting

Windows XP forces a reboot after installing the device

This problem can occur if an application is accessing a file while the New Hardware Wizard is trying to copy it. This usually occurs with the FTD2XX.DLL file. Selecting not to restart the computer then unplugging and re-plugging the device may allow the device to function properly without restarting. Restarting the machine will allow the device to work correctly.

Driver installation fails and Windows XP gives Error Code 10

Windows error code 10 indicates a hardware error or failed driver installation. This error may appear if a device has insufficient power to operate correctly (e.g. plugged into a bus powered hub with other devices), or may indicate a more serious hardware problem. Also, it may be indicative of USB root hub drivers being incorrectly installed. If the error persists, please contact the ARE-POC support department.

Windows XP displays an error and terminates installation

If the following screen is displayed with this message, Windows XP has been configured to block the installation of any drivers that are not WHQL certified.



Two options are available to successfully install the device. Either a certified version of the driver can be installed (if available) or the driver signing options can be changed to either warn or ignore to allow the installation to complete. To change the current driver signing setting, go to "Control Panel\System", click on the "Hardware" tab and then click "Driver Signing". The desired signing option may then be selected.

2.2 Installing USB2AD Drivers under Windows 7

To install drivers for USB2AD device under Windows 7, internet connection should be allowed prior connecting USB module to computer.

Connect the device to a spare USB port on your PC and system will automatically connect to the Windows Update website and install latest WHQL certified driver for your device.

If no access to the internet is available from your computer, follow the instructions in **Installation Guide for Windows 7** manual.

2.3 Important Notes

To avoid shock hazard, connect the USB cable only to a computer that has its power connector connected to a power receptacle with an Earth Safety Ground. Use care when plugging the module into a USB port on your computer. Make sure no cable is connected to the front panel of the DMM while you plug it into the USB port. Since the DMM is powered via a USB cable, it is important to make sure it gets the required 5V supply. Using the right USB cable is very important. Make sure this cable has a 24 AWG wires for power supply, indicated by marking on the cable such as <28/1P + 24/2C>. Be aware that there are a lot of cables which are marked <28/1P + 28/2C>. These have high resistance, and will not be adequate. Another issue can be with powered USB hubs. Some of the lower quality units can have upwords of 8V rather than the required 5V +/-5%. On initialization the DMM measures its internal supply voltage and returns an error or warning code if the power is inadequate.

3.0 Analog I/O Connector

Figure 3.1 shows the pin assignments of the USB2AD analog I/O connector. The connector is located on the front panel of the USB module. The analog I/O connector is a CANON 25-pin (DB25F) female connector.

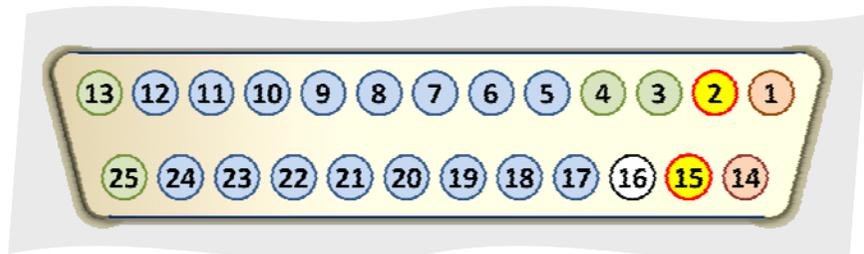


Figure 3.1 Pin assignments of the Analog I/O connector

Pin	Signal Description	Pin	Signal Description
1	I_OUT (Current output Lo)	14	I_OUT (Current output Hi)
2	+13V OUT (User power)	15	-12V OUT (User power)
3	AGND (Floating ground)	16	Reserved
4	AGND (Floating ground)		
5	0_IN Lo (Input channel 0)	17	0_IN Hi (Input channel 0)
6	1_IN Lo (Input channel 1)	18	1_IN Hi (Input channel 1)
7	2_IN Lo (Input channel 2)	19	2_IN Hi (Input channel 2)
8	3_IN Lo (Input channel 3)	20	3_IN Hi (Input channel 3)
9	4_IN Lo (Input channel 4)	21	4_IN Hi (Input channel 4)
10	5_IN Lo (Input channel 5)	22	5_IN Hi (Input channel 5)
11	6_IN Lo (Input channel 6)	23	6_IN Hi (Input channel 6)
12	7_IN Lo (Input channel 7)	24	7_IN Hi (Input channel 7)
13	AGND (Floating ground)	25	AGND (Floating ground)

3.1 Analog Inputs

The USB2AD differential analog inputs are software-configurable for bipolar or unipolar input ranges. Unipolar input means that the input voltage range is between 0 and V_{ref} where V_{ref} is a positive reference voltage. Bipolar input means that the input voltage range is between $-V_{ref}/2$ and $+V_{ref}/2$.

The measured input voltage must be connected between **Lo** and **Hi** terminals of the respective input channel. The terminal **Lo** must be connected also to a floating ground (**AGND** pin **3**, **4**, **13** or **25**). The signal 'grounding' is not necessary since a Hall probe, RTD, strain gauge bridge or other similar sensor is connected to analog output of the USB module.

The analog inputs have software-programmable gains of 1 to 512 allowing setting of analog input ranges from ± 9.8 mV to ± 5 V in bipolar mode, or from $+19.6$ mV to $+10$ V in unipolar mode. Amplified voltages are multiplexed into an analog-to-digital converter operating on the principle of modified Σ - Δ modulation of charge balancing.

The table shows actual unipolar and bipolar input ranges:

Gain	Unipolar Input Range	Bipolar Input Range
1	0 to +10 V	± 5.0 V
2	0 to +5.0 V	± 2.5 V
4	0 to +2.5 V	± 1.25 V
8	0 to +1.25 V	± 0.625 V
17	0 to +0.588 V	± 0.294 V
39	0 to +0.256 V	± 0.128 V
102	0 to +98 mV	± 49 mV
512	0 to +19.6 mV	± 9.8 mV

Integration time T_{IN} of analog inputs is software-programmable from 5 to 320 ms in four steps with corresponding resolution from 20 to 26 bits. Real resolution is limited by noise of the AD converter. For low gains (1x, 2x and 4x) the noise is limited by the AD converter itself and the achievable resolution is approx. 17, 18, 19, 20 bits for integration time 5, 20, 80 and 320 ms, respectively. In order to obtain total conversion time T_{CN} , 3 to 15 ms must be added (it depends on your PC performance) for data processing. Total conversion time of each channel for data processing versus effective analog input resolution is shown in the following table:

Resolution [bit]	20	22	24	26
Effective resolution [bit]	17	18	19	20
Integration time T_{IN} [ms]	5	20	80	320
Conversion time T_{CN} [ms]	12	26	86	326

For higher gains the resolution is limited by noise of the input amplifier, which is approximately 400 nV_{p-p} for $T_{IN} = 320$ ms and raises up to approx. 700 nV_{p-p} for $T_{IN} = 20$ ms. Corresponding resolution for gain 512 (range 0 - 19.6 mV) is 18 bits for integration time 320 ms.

Integration principle of conversion and galvanic isolation from PC guarantees higher immunity against spurious voltages. To reach high accuracy and stability, the USB module is equipped with auto calibration function which enables to eliminate ADC offset. Two options are available for auto calibration settings:

- calibration of a single input range - takes 3 integration periods ($3 \times T_{IN}$)
- calibration of all input ranges - takes $9 \times T_{IN}$

Calibration of the single input range is used in case that only one and the same range is used for all used inputs. Otherwise, calibration of all input ranges is recommended. It is possible to call auto calibration procedure periodically on long-running measurements.

3.2 Analog Output

The USB2AD module provides a programmable DC current source. The output current can be software-configured in the range of 0 - 100 mA (0 - 20 mA optional) with 16-bits output resolution. Analog output is connected to pins **14** (I_{OUT} current output **Hi**) and **1** (I_{OUT} current output **Lo**) of the CANON 25 connector.

It is possible to switch-off of the DC current source from an application software environment.

- ! Note, the current source provides $<1 \mu\text{A}$ cut-off current also in
- the switch-off state!

3.3 User Power Connections

Pins **2** ($+13V_{out}$) and **15** ($-12V_{out}$) of the CANON 25 connector supply voltage derived from internal circuit of the module. These $+13\text{V}$ and -12V voltages are referenced to the floating ground ($AGND$ pins **3**, **4**, **13** or **25**) and are available to power external accessories or circuitry. Maximum power rating for these power outputs is 10 mA.

- ! Be careful to use the $+13V_{out}$ and $-12V_{out}$ power outputs! Short-
- ing or grounding these outputs (pin **2** and/or pin **15**) of the analog I/O connector may damage internal circuitry of the module! AREPOC is not liable for damages resulting from such a connection.

3.4 Chopping on Analog Inputs

Chopping is a technique that can be used to cancel offset voltages errors that can arise at many points within a signal processing, e.g., the temperature dependent thermocouple voltages that arise when two dissimilar metals are joined. These offsets are generally undesirable and are particularly problematic if they change with temperature because a one-time zeroing is not sufficient to remove offset errors across all temperatures and supplies.

The solution implemented on USB2AD module is to chop the entire analog signal chain. In the first step, voltage at a given channel (differential analog input) is measured. Subsequently the input is alternately inverted (chopped) at the input multiplexer and voltage is measured again. This sequence may be repeated by setting number of chopping cycles in the *SetChop* procedure. The error (offset) voltage is removed by averaging of the measured voltages resulting from the chopping procedure.

The primary purpose of chopping on analog inputs is to eliminate offset and spurious thermocouple errors arising within different metals used for contacts like solder, wires, connector, pins metallization, etc. The typical application for Analog Inputs Chopping is thermocouple temperature measurement.

3.5 Chopping on Analog Output

Chopping technique of the USB2AD module is furthermore enhanced by possibility to choose between chopping on Analog Inputs or Analog Output. Implemented inverter of the analog output switches the bias direction of the current source. Procedure of the signal processing is the same as at the Analog Inputs Chopping. Advantage of using the output current inverter is elimination of spurious offset voltages generated in entire circuit of the sensor connection. The typical applications for Analog Outputs Chopping are biased sensors - like Hall probes, RTDs, strain gauges, etc.

The both chopping techniques are provided by firmware programmed in single-chip microcomputer of the USB module. User will set just type of modulation (input or output) and even-number of cycles in *SetChop* procedure. Since the Σ - Δ modulator needs to perform a conversion for each phase of the chopping, the output time from AD converter takes the number of conversion periods set in *pocvz* parameter of the *SetChop* procedure multiplied by integration time (*pocvz* \times T_{IN}). Relative good resolution/speed measurement ratio can be achieved by setting *pocvz* = 2 and T_{IN} = 20 ms.

4.0 Digital I/O Connector

Figure 4.1 shows the pin assignments for the USB2AD digital I/O connector. The connector is located on the front panel of the USB module. Before connecting or disconnecting of signal wires to the digital I/O connector the USB module must be unplugged from computer. The digital I/O connector is DIL16 male connector.

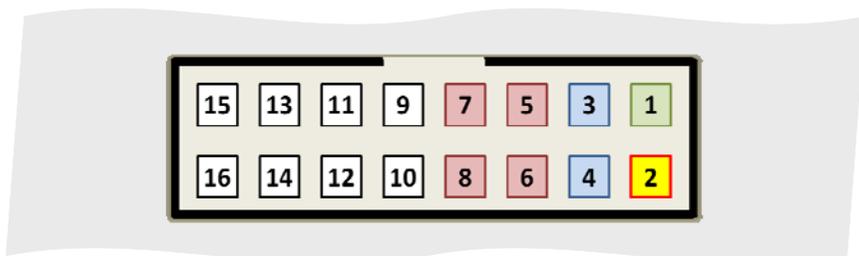


Figure 4.1 Pin assignments of the Digital I/O connector

Pin	Signal Description	Pin	Signal Description
1	DIG_GND (Ground)	2	+5V OUT (User power)
3	DIG_IN 0 (Input channel 0)	4	DIG_IN 1 (Input channel 1)
5	DIG_OUT 0 (Output channel 0)	6	DIG_OUT 1 (Output channel 1)
7	DIG_OUT 2 (Output channel 2)	8	DIG_OUT 3 (Output channel 3)
9	Reserved*	10	Reserved*
11	Reserved*	12	Reserved*
13	Reserved*	14	Reserved*
15	Reserved*	16	Reserved*

! *Pins (9 - 15) are reserved for firmware communication with factory programming interface. Do not connect any signal or power wires to these terminals! Alike, shorting or grounding of these terminals can damage internal circuitry of the USB module!

4.1 Digital Inputs and Outputs

The USB2AD provides 2 digital input and 4 digital output lines TTL/HCT compatible. A strobe signal is provided for latching the digital input/output signals into external circuitry. Digital I/O lines are programmable using *DigIn* and *DigOut* functions of the *ad24usb.dll* library.

The following specifications apply to the digital I/O lines. Absolute maximum voltage input rating is 5.0 V with respect to the DIG_GND.

Digital input specifications (referenced to DIG_GND):

- VIH input logic high voltage 2 V min.
- VIL input logic low voltage 0.8 V max.
- IIH input current load, logic high input voltage 20 μ A max.
- IIL input current load, logic low input voltage -20 μ A max.

Digital output specifications (referenced to DIG_GND):

- VOH output logic high voltage 2.4 V min.
- VOL output logic low voltage 0.5 V max.
- IOH output source current, logic high 2.6 mA max.
- IOH output sink current, logic low 24 mA max.

Maximum ratings are absolute ratings. Connections that exceed any of the max. ratings of input or output signals can damage the USB2AD module. AREPOC is NOT liable for any damages resulting from incorrect signal connections.

5.0 Cabling and Wiring Consideration

Environmental noise can affect the accuracy of measurements made with the USB module if you do not make proper considerations when running signal wires between signal sources and the USB2AD module. The following recommendations mainly apply to analog input signal routing to the module, although they are applicable for signal routing in general.

Use individually shielded twisted-pair wires to connect analog input signals to the USB2AD. With this type of wire, the signals attached to the inputs of channels 0 - 7 are twisted together and then covered with a shield. This shield is then connected at only one point to the signal source ground. This kind of connection is required for signals traveling through areas with large magnetic fields or high electromagnetic interference. To protect the signal lines from magnetic fields caused by electric motors, welding equipment, breakers or transformers, run the signal lines through special metal conduits. In making your own cabling, you may decide to shield your cables. The mating connector for the USB2AD analog I/O is a CANNON 25-pin male connector.

Figure 5.1 shows how to measure voltage outputs from several Hall probes connected in series, biased from the chopping current source of the USB module. In case, the Hall probes are biased by an external current source, at least one *Lo* input terminal must be connected to analog ground (AGND pin).

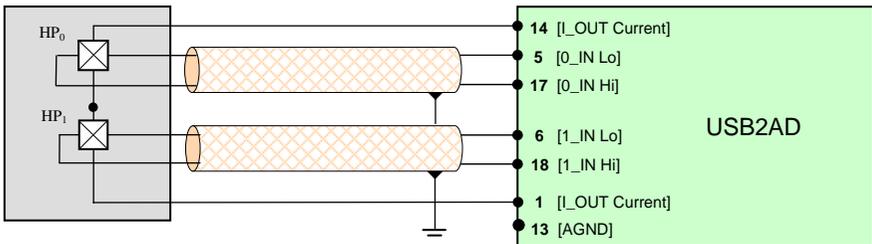


Figure 5.1 Connecting to Hall probes configuration

MULTIFUNCTION USB MODULE USB2AD

The same is valid for measurements with other sensors like RTDs, resistors, strain gauges, etc.

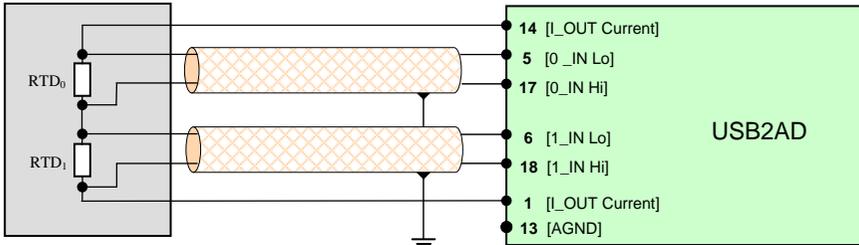


Figure 5.2 Connecting to RTDs configuration

Figure 5.2 shows how to connect two RTDs to the USB2AD module. It is possible to measure output of up to max. 8 RTDs connected in series to the module's inputs.

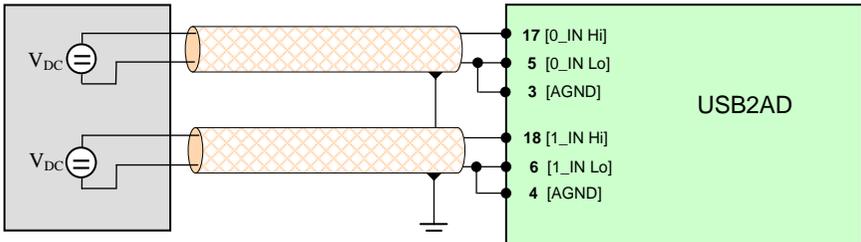


Figure 5.3 Connecting to floating voltage configuration

Figure 5.3 shows how to connect several DC floating signal sources to the USB2AD module. Note, **Lo** input terminal of each respective input must be connected to analog ground (AGND pin).

The input voltage from the **Lo** input or **Hi** input should not exceed the maximum input voltage (based on the analog ground of the module). If it exceeds the maximum voltage, the board may be damaged.

MULTIFUNCTION USB MODULE USB2AD

High resolution of the USB2AD module results from the application of auto-calibration method. This multifunction analog/digital I/O module can be used in applications where high accuracy, linearity and resolution are required. It can serve as a relatively low cost substitution of 6-1/2 decade system multimeter.

Technical Parameters:

Analog Inputs

Input resolution (programmable)	18 to 24 bits
Nonlinearity (if gain =1)	typ.: 0.0005 % of full scale max.: 0.0008 %
Differential nonlinearity	max.: 1 LSB
Max. error	0.005 % of reading 0.01 % of FS
Number of analog inputs	8 differential
Input range	0 - 10 V (unipolar) or ± 5 V (bipolar)
Programmable gain	1 - 128 (software selectable)
Input resistance	$> 100 \text{ M}\Omega$
Input bias current	typical: 10 nA, max.: 50 nA
Input overvoltage	$\pm 18\text{V}$ continues, powered $\pm 10\text{V}$ continues, unpowered
Typical noise	< 2 ppm of FS, or $< 0.5 \mu\text{Vp-p}$

Current Source

Output current range	0 - 100 mA (0 - 20 mA optional) DC
Output current resolution	16 bits (14 bits optional)
Output current ripple	< 0.0005 % of FS, or $< 0.5 \mu\text{A}$
Cut-off current	$< 1 \mu\text{A}$
Max. output voltage	7.8 V

Power Connections

Floating output voltage	+13 V, -12 V
Max. power load	10 mA

Digital Inputs and Outputs

4 digital outputs	TTL/HCT compatible
2 digital inputs	TTL/HCT compatible
Recommended warm-up time	> 15 min
Relative humidity	10% to 90% noncondensing
Overall dimensions	135 x 105 x 35 mm

Description of the AD24USB.DLL library

The USB2AD comes with functions in Dynamic Link Library *ad24usb.dll* which can be called from VC++, Delphi, Visual Basic or LabVIEW at *stdcall* convention. The USB2AD supports two modes of operation: *single conversion* or *periodical measurement*.

- In single conversion mode the USB2AD takes one sample immediately after calling function ***Adconv*** which returns measured value. An advantage of this mode is a simple programming. Disadvantage: triggering is derived from the PC so timing accuracy depends on host processor allocation.
- In periodical mode measurement sequence is programmed first using procedures ***ProgCh***, ***ProgEn***, ***Timer***, etc. Advantage of this mode is that triggering is controlled by internal clock of the USB2AD microprocessor and, therefore, the triggering is more accurate (PC will only read-out results before start of the next conversion). Disadvantage: during periodical measurement the auto-calibration is inaccessible; therefore, this mode is recommended for short-time measurement up to few minutes only.

The *ad24usb.dll* offers the following procedures and functions:

function *initUSB*:integer;

Returns number of available USB2AD modules connected to a system.

procedure *ReadSerialUSB*(adnum:integer; SerialNo:pointer);

Returns USB chip ID number at position **adnum**

function *InitSelectedAD*(num:integer):integer;

Initializes the selected USB2AD module.

num=0 is indexed to the 1st device connected to a system

Successful initialization returns **1**.

Value **0** means communication error with USB serial converter.

The function ***InitSelectedAD*** must be called at first, prior to any other function described below!

procedure SetRange(*typr:integer*);

This procedure sets unipolar or bipolar input range of the module. Value **typr=0** is for setting of unipolar and **1** for bipolar range.

The **SetRange** procedure must be called prior to **calibration** procedure!

procedure SetChop(*typchop, pocvz:integer*);

This procedure activates chopper.

Parameter **typchop** sets modulation mode: **0**—chopper is inactive, **1**—modulation of analog inputs, **2**—modulation of the analog output (current source).

Parameter **pocvz** sets number of cycles used for alternate switching. This value must be an even-number (**2, 4**, or more) for a proper value translation.

Prior calling the **SetChop** procedure, the **SetRange** procedure must be set to the bipolar range (**typr=1**)!

In case of the chopper activating, the **SetChop** procedure must be called prior to the **calibration** procedure!

procedure calibration(*typ, range, resol:integer*);

This procedure accomplishes auto-calibration of the USB2AD. Every used range with common resolution must be calibrated before measurement! Auto-calibration removes drift of inputs, so it is recommended to call this routine during measurement in interval tens of seconds.

Parameters:

typ - 0 - calibrate only selected range

1 - calibrate all ranges (call only if use more ranges during measurement)

range - 0...7 - select gain 1, 2, 4, 8, 17, 39, 102 or 512

resol - 0...3 - select integrating time 5, 20, 80 or 320 ms

function Adconv(*resol, range, chan:integer*):double

Returns value in Volts of the selected input channel.

Parameters:

resol, range - the same values as in the procedure **calibration**

chan - 0...7 - selects the measured input channel number

Prior to setting the **Adconv** parameters, respective **resol** and **range** values must be pre-calibrated by calling the **calibration** procedure!

procedure ProgEn(chan, resol:integer);

Sets number of measured channels and resolution in the periodical mode.

Parameters:

chan - number of measured channels

resol - see procedure **calibration**

procedure ProgCh(chan, range:integer);

Sets channel and range for every selected input.

Parameters - see function **Adconv**

procedure Timer(burst:byte; time:word);

Sets total measurement period in periodical measurement.

Parameters:

burst - always **1** - second channel is measured immediately after the 1st, 3rd channel after 2nd, etc.

time - total measurement period in ms, max. value is 65000.

For a proper function of this procedure the **time** value must be equal or bigger then the total sum of conversion time of all measured channels.

procedure StartAD;

Starts measurement in periodical mode.

procedure StopAD;

Stops measurement in periodical mode.

function GetSample(range:integer):double

Returns measured voltage value (in Volts) in periodical mode.

range - see procedure **calibration**

function GetDACresolution:integer;

Returns resolution of the installed DA converter (14 or 16 bits).

procedure DAC(value:word); stdcall; export;

Sets value of the current output (I_OUT) of the current source.

Parameters:

value - **0...16000** for 14-bit DA converter

0...64000 for 16-bit DA converter

The **value** range corresponds to the output current range of 0...100 mA (optional to current range of 0...20 mA).

MULTIFUNCTION USB MODULE USB2AD

function GetCurrentRange:integer;

Returns value of the range of the current source (20 or 100) mA.

procedure DigOut(value:byte);

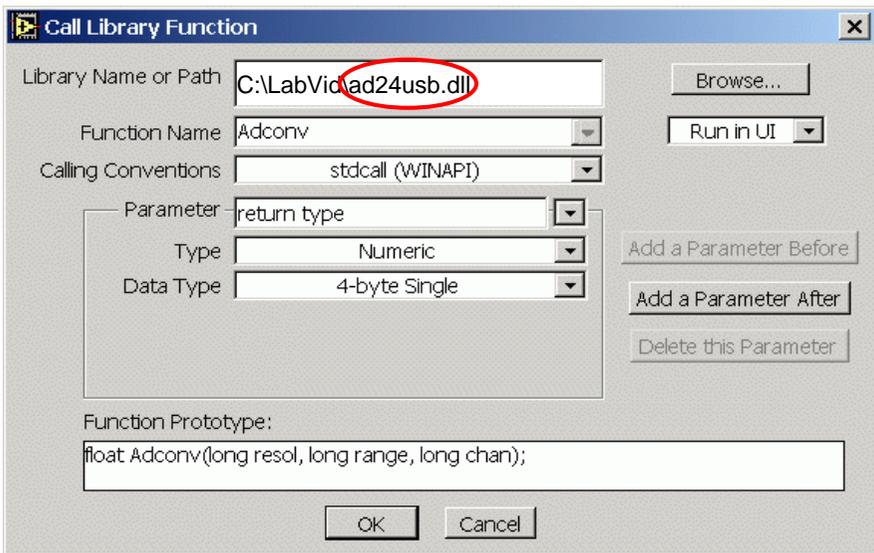
Writes value on a digital output.

function DigIn:byte;

Reads value from a digital input.

Using the LabVIEW™

Procedures and functions of the *ad24usb.dll* can be called also from the LabVIEW environment in Windows operating system. The calling convention is *stdcall*; a simple example application can be found in the file **USB2AD.vi** stored on installation the *CD* supplied with the module. Before the first run it is necessary to set proper path to the *ad24usb.dll*. After loading the **USB2AD.vi** switch to the diagram, double click on every function and set proper name and path to the library.



Note, library *ad24usb.dll* must be installed in your application directory, or in the LabVIEW directory.

Programming Example

Here is a simple common example programmed in Delphi which can be simple adapted to any programming language.

Single conversion example:

```
if InitSelectedAD(0) = 0 then //USB module initialization
begin
    MessageBox(0, 'Unable to connect to USB2AD, 'Error', MB_OK);
    Exit;
end;
unip:= SetRange; //set unip/bip range, unip=1-bipolar
resol:=2; //Tint=80 ms; resolution cca 19 bits}
Calibration(0,0,resol); //Calibration - only range ±5V
u:= ADConv(resol,0,1); //channel 1, range ±5V
```

Periodical measurement example:

```
if InitSelectedAD(0) = 0 then //USB module initialization
begin
    MessageBox(0, 'Unable to connect to USB2AD, 'Error', MB_OK);
    Exit;
end;
unip:= SetRange; //set unip/bip range, unip=1-bipolar
resol:=2; //Tint=80 ms; resolution cca 19 bits}
Calibration(1,0,resol); //Calibration - all ranges
ProgEn(3,resol); //Measure 3 channels
ProgCh(0,0); //Input 0, range ±5V
ProgCh(1,2); //Input 1, range ±1.25V}
ProgCh(2,1); //Input 2, range ±2.5V}
Timer(1,1000); //Burst mode, period 1000 ms}
StartAD;
for i:=1 to 5 do
begin
    ux[0,i]:=GetSample(0); //read input 0, range 0
    ux[1,i]:=GetSample(2); //Input 1, range 2
    ux[2,i]:=GetSample(1); //Input 2, range 1
end;
StopAD;
```

Setting the current example:

```
DAC(4000); //current 6.25 mA (for 16bit 0-100 mA current surce)
```

MULTIFUNCTION USB MODULE USB2AD

Warranty

The USB2AD multifunction module is warranted against defects in materials and workmanship for a period of one year from the date of shipment, as evidenced by receipts or other documentation. Arepoc will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor. A Return Material Authorization number must be obtained from the factory and clearly marked on the outside of the package before the module will be accepted for warranty work. Arepoc believes that the information in this document is accurate. In the event that technical or typographical errors exist, we reserve the right to make changes to subsequent editions of this document without prior notice to holders of this edition. In no event shall Arepoc be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, AREPOC MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF AREPOC SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. AREPOC WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of Arepoc will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against Arepoc must be brought within one year after the cause of action accrues. Arepoc shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the Arepoc installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Warning Regarding Use of Arepoc Products

AREPOC PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN. IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM AREPOC' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE AREPOC PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY AREPOC, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF AREPOC PRODUCTS WHENEVER AREPOC PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Information furnished is believed to be accurate and reliable. However, AREPOC s.r.o. assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties, which may result from its use. Product and company names are trademarks or trade names of their respective companies. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied.

AREPOC s.r.o., Iljušinova 4, 851 01 Bratislava, SLOVAKIA
FAX: +421 2 6382 4613 E-mail: **arepoc@zmail.sk**
<http://www.arepoc.sk>