

**24 bitový AD Ethernet modul**

**AD24ETH**

**Návod k použití**

# Obsah

1. Úvod.....	3
2. Technické parametry.....	3
A/D převodník: .....	3
Zdroj proudu: .....	4
Digitální vstupy/výstupy: .....	4
3. Instalace modulu.....	5
3.1. Zapojení vst. konektoru CANON25 zásuvka.....	7
3.2. Zapojení konektoru dig. IO DIL16.....	8
4. Popis činnosti modulu.....	9
4.1. AD převod .....	9
4.2. Střídavá modulace vstupu.....	10
4.3. Režimy měření.....	11
5. Programování modulu.....	12
5.1. Instalace knihovny ovladače modulu.....	12
5.2. Inicializace a odpojení modulu.....	12
5.3. Popis AD funkcí knihovny ad24eth.dll.....	13
5.4. Popis DA funkcí knihovny ad24eth.dll .....	15
5.5. Popis dig IO funkcí knihovny ad24eth.dll .....	16

# 1. Úvod

AD24ETH je USB měřicí modul, který obsahuje autonomním procesorem řízený AD převodník s vysokou rozlišovací schopností, který je galvanicky oddělen od obvodů počítače. AD převodník pracuje na principu modifikované metody vyrovnávání náboje používané u multimetrů vyšších tříd přesnosti. Integrační princip převodu a galvanické oddělení zajišťují vysokou odolnost vůči rušivým napětím, modul je ideální pro měření malých signálů např. z různých čidel.

Vysoká přesnost AD24ETH je dána využitím metody autokalibrace, modul je možno použít všude tam, kde jsou kladeny vysoké nároky na přesnost, linearitu a rozlišovací schopnost. Modul svými parametry může nahradit až 6-ti dekádové multimetry při podstatně nižší ceně.

## 2. Technické parametry

**Napájení** : 5 V DC z externího zdroje (je součástí dodávky), odběr bez proudového zdroje 500 mA, s proudovým zdrojem max. 750 mA.

**Komunikace**: Ethernet rozhraní 100/10 Mbit/s, TCP IP protokol, zařízení funguje jako server.

### **A/D převodník:**

- 8 diferenciálních nebo 16 SE galvanicky oddělených vstupů
- základní rozsah 0 až 10 V unipolárně nebo +- 5 V bipolárně, programově volitelné zesílení 1 až 128, (1,2,4,8,17,39,102,512 pro diferenciální verzi)
- programovatelné rozlišení 18 až 24 bitů (16 bitů u AD16ETH)
- rychlost převodu 60 převodů/s při rozlišení 18 bitů, 3 převody/s - 24 bitů
- chyba linearity typ. 0.0002%, max. 0.0005% (max. 0.0025% pro AD16ETH)

- šum menší než 0.0001% z rozsahu, popřípadě < 500 nV š-š (0.0016% AD16ETH)
- max. chyba 0.005% z údaje + 0.001% z rozsahu + 5 uV
- max. chyba 0.008% z údaje + 0.002% z rozsahu + 5 uV pro AD16ETH

***Zdroj proudu:***

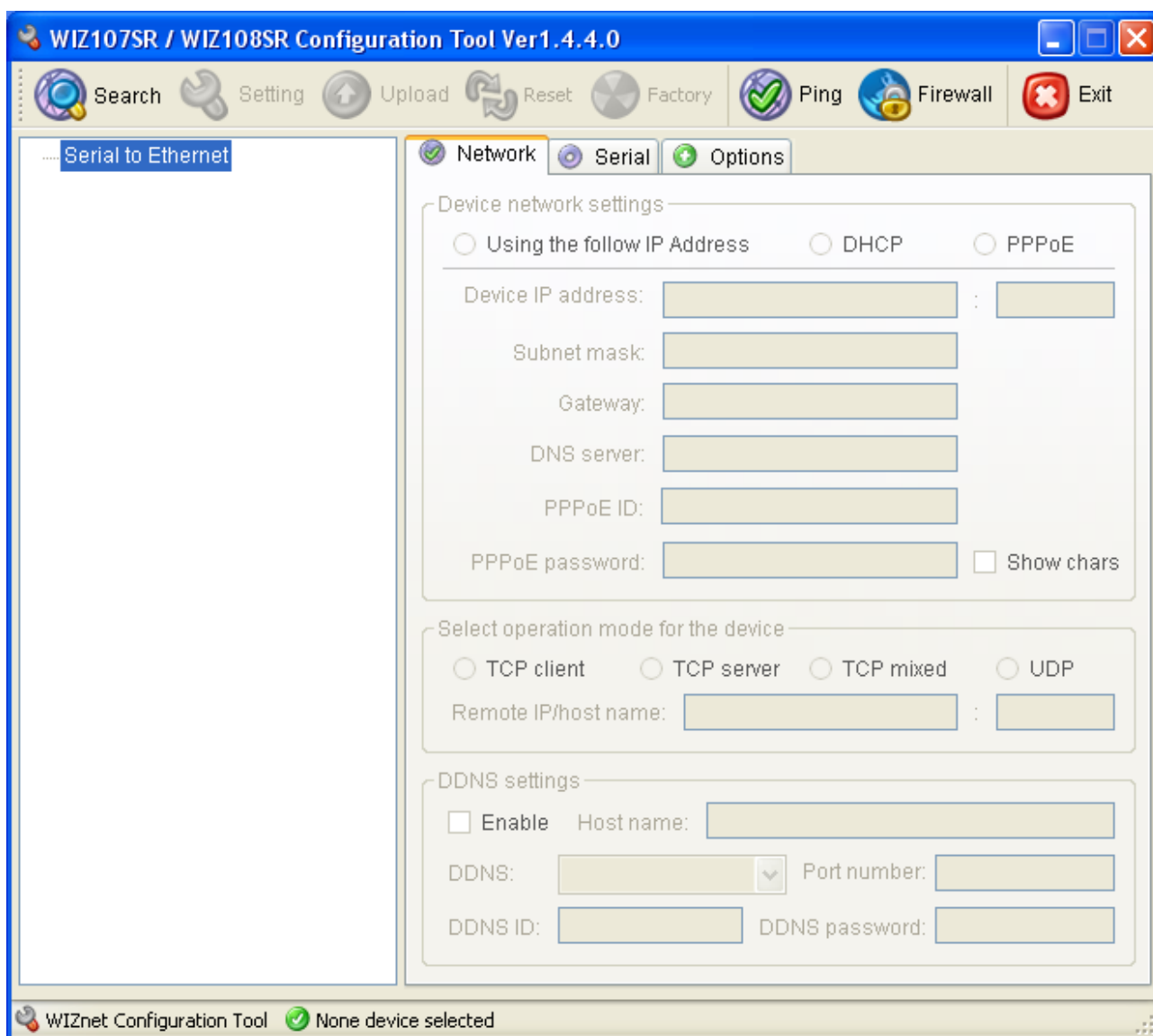
- Programovatelný rozsah 0 až 20 mA nebo 100 mA, po dohodě i jiný nebo napětí 0 až 5V nebo 0 až 10 V.
- Rozlišení 16 bitů
- Šum < 5 ppm

***Digitální vstupy/výstupy:***

- 4 digitální výstupy TTL kompatibilní
- 2 digitální vstupy TTL kompatibilní

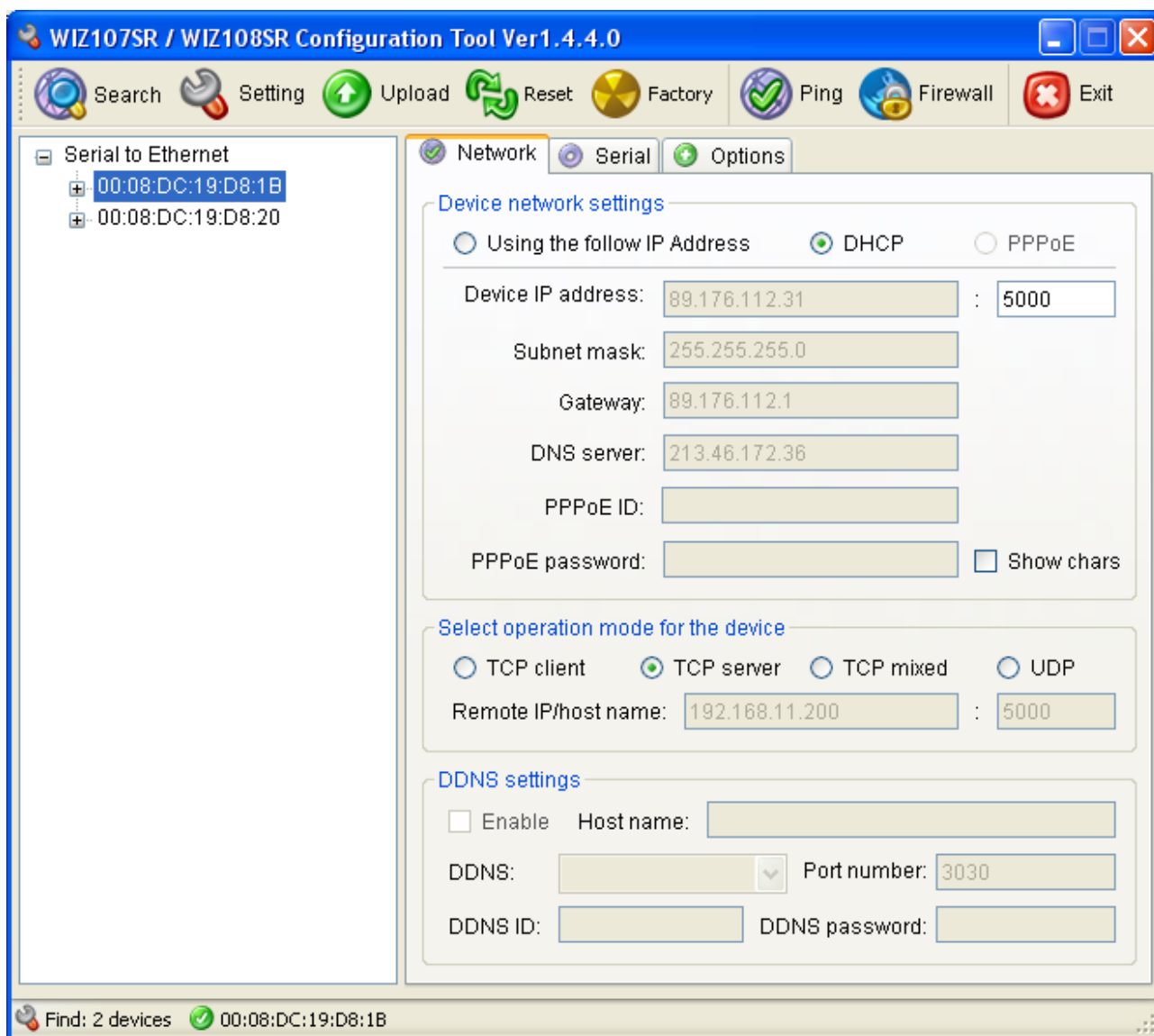
### 3. Instalace modulu

Modul připojíme k napájení pomocí dodaného zdroje a k síti pomocí Ethernet kabelu. Modul AD24ETH používá ke komunikaci přes Ethernet komunikační modul WIZ107SR. Výrobce modulu dodává k jeho konfiguraci program **ConfigTool107**, který se nachází v adresáři Wiznet na přiloženém CD. Tento program umožňuje snadné vyhledání modulu připojeného k LAN a zjištění a nastavení IP adresy. Znalost adresy je nutná pro komunikaci s modulem. Úvodní obrazovka programu vypadá takto:



Nejprve zvolíme tlačítko Search a zvolíme způsob vyhledávání – v LAN síti pomocí *UDP broadcast*, ve WAN síti pomocí *TCP unicast* – podrobnosti o

připojení do WAN sítě jsou v souboru **WIZ107SR\_AN\_Public\_IP.pdf**. Pokud je do sítě připojen alespoň jeden modul se zapnutým napájením a je úspěšně nalezen, zobrazí se následující okno:



Podle MAC adresy zvolíme modul, jehož adresu chceme zjistit nebo změnit. Výchozí nastavení modulu používá pro nastavení IP adresy DHCP server. Pokud chceme zvolit vlastní IP adresu, zvolíme *Using the follow IP address*. Přidělenou nebo nastavenou adresu si poznamenáme a pokud došlo k nějaké změně, nahrajeme novou adresu do modulu pomocí tlačítka *Setting*. Parametry na záložce *Serial* a *Options* neměňte, komunikace s modulem by nefungovala! Základní Ethernet komunikaci je možno ověřit pomocí tlačítka *Ping*.

**Doporučený postup nastavení:** Pokud neznáme předem použitelnou volnou IP adresu, je vhodné nejprve adresu nechat přidělit pomocí DHCP serveru (pokud v dané síti existuje), poté přepnout na pevnou IP adresu a nahrát ji zpět do modulu. Poté je vhodné se připojit k DHCP serveru a zvolenou adresu zařadit mezi vyhrazené. Tím zajistíme, že při změně konfigurace sítě server nepřidělí zvolenou adresu jinému zařízení.

Popis instalace dalších programů a případné řešení potíží viz soubor **readme** na CD.

### ***3.1. Zapojení vst. konektoru CANON25 zásuvka***

- 1 – IOOUT – DA výstup LO svorka
  - 14 – IOOUT – DA výstup Hi svorka
- 2 - +15V izol
  - 15 - -15V izol
- 3,4 – izol GND
  - 16 – čidlo teploty pro termočlávkovou svorkovnici
  - 17 – Hi IN0
- 5 – Lo IN0 (IN8 pro SE verzi)
  - 18 – Hi IN1
- 6 – Lo IN1 (IN9 pro SE verzi)
  - 19 – Hi IN2
- 7 – Lo IN2 (IN10 pro SE verzi)
  - 20 – Hi IN3
- 8 – Lo IN3 (IN11 pro SE verzi)
  - 21 – Hi IN4
- 9 – Lo IN4 (IN12 pro SE verzi)
  - 22 – Hi IN5
- 10 – Lo IN5 (IN13 pro SE verzi)
  - 23 – Hi IN6
- 11 – Lo IN6 (IN14 pro SE verzi)
  - 24 – Hi IN7
- 12 – Lo IN7 (IN15 pro SE verzi)
- 13,25 – izol GND

**Zapojení pro diferenciální verzi modulu:** Měřené napětí se připojuje mezi svorky Lo a Hi zvoleného vstupu. Svorku Lo je nutno spojit s plovoucí zemí

-svorky 3,4,13,25 (Kromě výjimečných případů jako např. měření tenzometrů, odporových čidel teploty, HALL čidel apod.).

**Zapojení pro SE verze:** Měřené napětí se připojuje mezi zvolený vstup a plovoucí zem pin 3,4,13,25. Tuto plovoucí zem je možno dle potřeby uzemnit.

Na svorky 3 a 15 je vyvedeno přes propojky izolované napájení  $\pm 12$  V. Tímto napětím je možno napájet ev. další obvody. Maximální povolený odběr je 10 mA. **Pozor!** Při zkratu těchto napájení přestane modul fungovat!

### ***3.2. Zapojení konektoru dig. IO DIL16***

1 – GND                      2 – Napájení 5 V

3 – IN0                        4 – IN1

5 – OUT0                      6 – OUT1

7 – OUT2                      8 – OUT3

9,10,11,12,13,14,15,16 Interní signály pro připojení expanzních desek - nesahat!



## 4. Popis činnosti modulu

### 4.1. AD převod

Jádrem modulu je integrační AD převodník řízený samostatným mikroprocesorem. Volbou doby integrace je možno volit dobu převodu a dosažené rozlišení AD převodníku. Jednotlivé možnosti jsou v tab.1.

Doba integrace [ms]	Doba převodu [ms]	Rozlišení bit
5	6 (22)	22
20	21 (37)	24
80	81 (97)	26
320	321 (337)	28

Tab.1.

Celková doba převodu záleží i na způsobu komunikace s modulem. Pokud je převod řízen vnitřním časovačem, modul nemusí čekat na přenos dat po Ethernetu a reakci PC, pak je možno dosáhnout kratších časů. Pokud je každý převod spouštěn jednotlivě, pak se uplatní i doba přenosu po sběrnici, doba reakce PC a platí orientačně časy uvedené v závorce.

Rozlišení uvedené v tabulce je pouze „hrubé“ rozlišení vlastního AD převodníku, využitelné rozlišení modulu je nižší a je dáno vlastním šumem modulu. Šum modulu v uVšš pro jednotlivá rozlišení a zesílení je v tab.2. Šum byl měřen pro unipolární rozsah a doba měření byla vždy 10 s.

Tint zesílení	1	2	4	8	16	32	64	128
<b>5 ms</b>	41	23	13	5	2.9	1.5	0.8	0.5
<b>20 ms</b>	10.9	6.5	3.8	2.0	1.0	0.65	0.37	0.26
<b>80 ms</b>	2.6	2.2	1.9	1.1	0.6	0.40	0.16	0.14
<b>320 ms</b>	1.3	1.2	1.0	0.5	0.3	0.25	0.18	0.12

Tab. 2.

Pozor! Vzhledem k náhodnému charakteru šumu a udávané mezivrcholové hodnotě je rozptyl hodnot i při opakovaném měření téhož modulu až 50%, pro nejnižší hodnoty až 100%! Z tabulky je patrné, že pro malá zesílení je šum určen vlastním šumem AD převodníku, pro větší zesílení se šum příliš nemění a

je určen šumem vstupního zesilovače. Šum a ofset vst. části u diferenciální verze je možno výrazně redukovat pomocí střídavé modulace popsané dále. Výše uvedené údaje platí pro AD24ETH, AD16ETH má rozlišení 16 bitů bez ohledu na dobu integrace.

Pro dosažení vysoké přesnosti je modul vybaven funkcí autokalibrace, při které se změří vlastní ofset a rozsah modulu. Je možno volit buď kalibraci jediného rozsahu nebo kalibraci všech rozsahů. Kalibrace jediného rozsahu trvá 3 doby převodu, kalibrace všech rozsahů trvá 9 dob převodu. Před měřením na kterémkoliv rozsahu je nutno kartu se zadaným rozlišením nejprve okalibrovat! Pokud měříme na jediném rozsahu, postačí kalibrace tohoto rozsahu, jinak je vhodnější použití kalibrace všech rozsahů. Autokalibraci je vhodné opakovat i v průběhu delších měření s periodou cca desítky sec. Pro kratší měření, u kterých je prioritou co nejnižší šum je naopak vhodnější provést autokalibraci pouze před startem měření.

## ***4.2. Střídavá modulace vstupu***

U verze modulu s diferenciálními vstupy lze výrazně snížit ofset i šum karty na nejmenších rozsazích pomocí střídavé modulace vstupu. Tato modulace pracuje následovně: Nejprve je pro daný vstup změřeno napětí přímo a poté s prohozenými vstupy a obě napětí jsou od sebe odečtena a vydělena dvěma. Tento postup je opakován se zadaným počtem opakování. Výsledkem je výrazná redukce ofsetu a nízkofrekvenčního šumu typu  $1/f$ . Doporučená doba integrace pro jeden převod je 20 ms pro potlačení rušení 50 Hz a jeho násobků, počet opakování závisí na požadované výsledné periodě měření. Výsledný šum klesá s odmocninou z počtu měření, takže např. pro čtyřnásobné zvýšení počtu měření klesne šum 2 x.

Tento velmi účinný princip eliminace ofsetu, šumu a parazitních termoelektrických napětí je dále vylepšen možností střídavé modulace budicího napětí nebo proudu při použití různých čidel jako např. tenzometrického můstku

nebo odporových čidel teploty, HALL snímačů apod. Díky modulaci přímo budicího napětí jsou eliminována i parazitní termoelektrická napětí generovaná na přívodech k čidlu.

Oba režimy modulace zajišťuje firmware modulu, uživatel pouze zvolí typ modulace a počet opakování.

### ***4.3. Režimy měření***

modul může pracovat ve dvou režimech: **jednorázovém** a **periodickém**.

**Jednorázový** režim je jednodušší – při každém zavolání funkce *Adconv* s příslušnými parametry se provede jeden odměr a funkce vrátí přímo hodnotu změřeného napětí. Pokud byla aktivována střídavá modulace, je proveden určený počet měření a vrácen celkový výsledek. Tento režim je vhodný pro většinu úloh, kdy nezáleží na naprosto přesném časování okamžiku převodu. Mezi volání funkce *Adconv* je možno vkládat i kalibrace převodníku.

**Periodický** režim je složitější, po nastavení parametrů převodu je modul spouštěn vlastním časovačem s nastavenou periodou měření. Tento režim je vhodný pro krátkodobější měření – desítky až stovky sec, u kterých je nutno přesněji definovat okamžik převodu a nevádí nemožnost autokalibrace v průběhu měření. Díky tomu, že naměřená data jsou předávána na pozadí i v průběhu měření, neuplatní se latence Ethernetu ani doba zpracování dat v PC a dosažitelná vzorkovací perioda je vyšší než u jednorázového měření.

## 5. Programování modulu

### 5.1. Instalace knihovny ovladače modulu

Instalace knihovny je velmi jednoduchá a spočívá v nakopírování knihovny **ad24eth.dll** do pracovního adresáře aplikace. Příklady použití viz dodané demo programy na CD. Všechny funkce jsou vytvořeny s volací konvencí *stdcall* a jsou volatelné z Delphi, C,C++, VB, LabView.

### 5.2. Inicializace a odpojení modulu

***function ConnectAD(modul:integer;IPaddress:pchar;port:integer) :integer***

Parametry: **modul** – číslo modulu, které bude použito při veškeré následné komunikaci s modulem, povolený rozsah 0 až 7, **Ipaddress** - IP adresa modulu nastavená pomocí programu **ConfigTool107** (viz kap. 3). **port** – číslo portu, v LAN síti vždy 5000, ve WAN síti při přesměrování portu i jiné.

Tato funkce inicializuje driver a modul a otevře socket spojení s modulem, vrací 1 pokud je úspěšná, hodnota 0 znamená, že Ethernet komunikace funguje ale vlastní modul neodpovídá. Hodnota větší než 10000 určuje číslo chyby při navazování komunikace s Ethernet částí, nejčastěji je chybná IP adresa. Funkci je nutno ji volat před jakoukoliv další prací s modulem.

***DisconnectAD(modul:integer)***

tato funkce ukončí spojení s daným modulem a uvolní ovladač z paměti.

### ***5.3. Popis AD funkcí knihovny ad24eth.dll***

#### ***procedure SetRange(modul,typr:integer)***

tato procedura nastavuje unipolární nebo bipolární rozsah vstupu modulu, vstupní rozsah musí být nastaven před zavoláním funkce ***Calibration!*** **typr=0** – znamená unipolární rozsah, 1 – bipolární.

#### ***procedure SetChop(modul,typchop,pocvz:integer)***

tato procedura aktivuje střídavou modulaci („chopper“) pro odstranění vst. ofsetu a driftu u diferenciální verze karty, princip viz. *Kap 4.2*. Parametr **typchop** určuje typ modulace – 0 – vypnuto, 1 – modulace vstupu, 2 - modulace výstupu. Modulace výstupu je možná pouze s příslušným hardware pro buzení čidla! Parametr **pocvz** určuje počet vzorků použitých pro výpočet, musí to být sudé číslo, zároveň určuje celkovou dobu převodu. Celková doba převodu je dána součinem doby jednoho převodu (typicky 21 ms) a počtu vzorků použitých pro modulaci. Pokud je aktivována střídavá modulace, pak funkce ***Adconv*** provede celkem **pocvz** převodů a vrátí celkový výsledek. Při aktivaci modulace je vhodné používat dobu integrace 20 ms (**resol = 1**). Vzhledem k principu modulace je nutno vždy nastavit bipolární rozsah!

#### ***procedure Calibration(modul,typ,range,resol:integer);***

tato procedura zkalibruje jeden nebo všechny rozsahy s daným rozlišením. Kalibraci je nutno volat až po proceduře ***SetRange*** a případné aktivaci střídavé modulace.

Má tyto parametry:

**typ** – 0 – jeden rozsah – používá se pokud měříme na jediném rozsahu, 1 – všechny rozsahy – trvá cca 3 x déle, pokud měříme na různých kanálech s různými rozsahy.

**range** – pokud je typ = 0, pak určuje, který rozsah je kalibrován, 0 .. 7 - 0 – zesílení 1, rozsah +-5 V bip nebo 0-10 V unip...7 – zesílení 128 tj rozsah +-38mV nebo 0-76 mV pro SE verze, u diferenciální verze jsou zesílení 1,2,4,8,17,39,102,512.

**resol** - rozlišení 0 až 3 nastavuje dobu integrace 5,20,80,320 ms a odpovídající rozlišení cca 22 až 28 bitů (16 bitů pro AD16ETH).

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

tato funkce vrací napětí ve V na vstupu **chan** – 0 až 7 (0 až 15 pro SE verze), parametry **resol,range** viz procedura **calibration**. Příslušný rozsah a rozlišení musí být předem zkalibrováno voláním procedury **calibration**!

***procedure StartAdconv(modul,resol,range,chan:integer)***

tato procedura je první částí funkce **Adconv**, spustí měření ale nečeká na výsledek převodu.

***function GetAdconv(modul:integer):double***

tato funkce převezme výsledek převodu zahájený **StartAdconv**. Použití těchto funkcí má smysl, pokud je pro měření použito více modulů – nejprve odstartujeme měření na více modulech postupným voláním **StartAdconv** a poté postupně přebíráme naměřené hodnoty. Při měření s jedním modulem je jednodušší použít rovnou **Adconv**.

**Funkce pro periodický režim:**

***procedure ProgEn(modul,chan,resol: integer);***

Tato procedura určuje počet měřených kanálů a jejich rozlišení, po jejím zavolání následuje příslušný počet volání **ProgCh**. Má parametry **chan, resol**, jejich hodnoty viz výše.

***procedure ProgCh(modul,chan,range: integer);***

Tato procedura nastaví pro každý aktivovaný kanál příslušné zesílení, význam parametrů viz výše.

***procedure Timer(modul:integer;burst:byte;time:word);***

Tato procedura nastaví periodu měření, tato doba musí být delší než součet dob převodu všech aktivních kanálů. Parametry: ***burst*** – vždy 1 – jednotlivé kanály jsou měřeny ihned po sobě, ***time*** – perioda měření v ms, max 65000.

***procedure StartAD(modul:integer);***

Tato procedura spustí převod se zadanými parametry.

***procedure StopAD(modul:integer);***

Tato procedura ukončí převod.

***function GetSample(modul,range:integer):double;***

Tato funkce vrací naměřenou hodnotu napětí ve V.

#### ***5.4. Popis DA funkcí knihovny ad24eth.dll***

***function GetDACresolution(modul:integer):integer;***

Tato funkce vrací rozlišení použitého DA převodníku – 16 bitů

***function GetCurrentRange(modul:integer):integer;***

Tato funkce vrací použitý proudový rozsah 20 nebo 100 mA

***procedure DAC(modul:integer;hodnota:word);stdcall; export;***

Tato procedura zapíše na DA výstup danou hodnotu, pro 16 bitů max 64000.

### ***5.5. Popis dig IO funkcí knihovny ad24eth.dll***

***procedure DigOut(modul:integer;value:byte);***

Tato procedura zapíše na dig. výstup danou hodnotu.

***function DigIn(modul:integer):byte;***

Tato funkce přečte hodnotu z číslicového portu.