Komunikace v síti ARION

Abstrakt

Realizace komunikační sítě rozšiřujících V/V modulů.

Autor: Václav Kaczmarczyk, Jiří Palát
Dokument: ap0005_cz_02.pdf

Příloha

Obsah souboru: ap0005_cz_02.zip

arion_p1_cz_01.dso	Příklad č. 1 – Periodická komunikace s V/V moduly
arion_p2_cz_02.dso	Příklad č. 2 – Různá perioda komunikace
arion_p3_cz_01.dso	Příklad č. 3 – Komunikace s ovladačem **NOA21**
arion_p4_cz_01.dso	Příklad č. 4 – Komunikace s **DM-UI8DO8** a **DM-UI8RDO8**

Copyright (c) 2009, AMiT®, spol. s r.o.

Obsah

Historie revizí 3
Související dokumentace 3
1. Definice použitých pojmů 4
1.1. Stará a nová koncepce knihovny ARION 4
2. Protokol ARION 5
2.1. Komunikační režimy protokolu ARION 5
3. HW rozšiřujících modulů 6
3.1. HW konfigurace rozšiřujících modulů 6
3.2. Význam LED 7
4. Realizace sítě ARION 8
4.1. Zapojení komunikační sítě 8
5. Časové poměry 9
5.1. Doba inicializace sítě 9
5.1.1 Doba inicializace úplné sítě 9
5.1.2 Doba inicializace neúplné sítě 10
5.2. Výpočet minimální periody komunikace s moduly 10
5.3. Doba první odezvy sítě 11
5.4. Detekce ztráty spojení (Guardtime) 11
6. SW konfigurace 12
6.1. Programová obsluha 12
6.1.1 Inicializace sítě 12
6.2. Periodická komunikace s rozšiřujícími moduly 13
6.3. Popis modulů pro komunikaci 13
7. Ukázkové aplikace 16
7.1. Příklad č. 1 – Periodická komunikace s V/V moduly 16
7.2. Příklad č. 2 – Různá perioda komunikace 18
7.3. Příklad č. 3 – Komunikace s ovladačem NOA21 20
7.4. Příklad č. 4 – Komunikace s DM-UI8DO8 a DM-UI8RDO8 21
8. Nejčastější problémy 23
8.1. Nedaří se navázat komunikaci 23
8.2. Komunikace je navázána, ale nefunguje spolehlivě 23
9. Dodatek A – Produkty firmy AMiT 24
9.1. Dostupné rozšiřující moduly firmy AMiT pro ARION 24
9.2. Využití funkce čítačových vstupů modulů DM-DI24 24
9.3. Použití modulu DM-PDO6Ni6 25
9.4. Nástěnné ovladače NOAxx 26
9.4.1 Popis I/O struktury ovladačů NOA2x 27
10. Dodatek B - Produkty jiných výrobců 28
11. Technická podpora 29
12. Upozornění 30

Historie revizí

Verze	Datum	Změny
001	13. 2. 2008	Nový dokument
002	6. 5. 2009	Oprava rozsahu u ARN_AI a ARN_AO. Přidány kapitoly 1.1, 7.3 a 7.4. Vytvořeny příklady č. 3 a 4.

Související dokumentace

1) Nápověda k vývojovému prostředí DetStudio
2) AP0002 – Komunikace v síti MP-BUS
   soubor: ap0002_cz_xx.pdf
3) AP0016 – Zásady používání RS485
   soubor: ap0016_cz_xx.pdf
4) AP0021 – Komunikace v síti 868 MHz
   soubor: ap0021_cz_xx.pdf
5) AP0025 – Komunikace v síti ARION – definice tabulkou
   soubor: ap0025_cz_xx.pdf
6) AP0028 – Zařízení OpenTherm v síti ARION
   soubor: ap0028_cz_xx.pdf
7) Katalogové listy k modulům **DM-xxx**
   soubory: dm-xxx_d_cz_xxx.pdf
8) Katalogový list k ovladačům **NOAxx**
   soubor: noaxx_d_cz_xxx.pdf
9) Návod – popis protokolu ARION
   soubor: arion_ms_cz_xxx.pdf

1. Definice použitých pojmů

Moduly DM-xxx
Moduly umožňující, prostřednictvím komunikační sítě ARION, rozšířit počet vstupů a výstupů řídicího systému.

Ovladače NOAxx
Nástěnné ovladače, které slouží k měření teploty prostoru, nastavení požadavku korekce teploty, nastavení režimů a jiných (dle typu ovladače), komunikujících v síti ARION.

DetStudio
Vývojové prostředí firmy AMiT, které slouží pro parametrizaci řídicích systémů. Toto prostředí je volně ke stažení na www.amit.cz.

RS485
Je poloduplexní sériová sběrnice umožňující komunikaci více jednotek na jednom signálovém páru. Maximální počet připojených jednotek na jednom segmentu sběrnice je 32 (max. počet jednotek na síti ARION je 63). Maximální přenosová rychlost na síti ARION je 57600 Bd. Více informací nalezete v dokumentu AP0016 – Zásady používání RS485.

1.1. Stará a nová koncepce knihovny ARION

V této aplikační poznámce je uveden popis SW konfigurace pomocí modulů ARN_xxx odpovídajících staré koncepci knihovny ARION. Tyto moduly jsou v aktuální verzi knihovny ARION nadále podporovány (tj. staré aplikace používající moduly ARN_xxx budou fungovat bez úprav), ale jejich repertoár ani funkčnost nebude nadále rozšiřována. Proto doporučujeme použití nových modulů ARI_xxx s možností definice rozšiřujících modulů ARION pomocí tabulky. Veškeré informace týkající se nové knihovny ARION naleznete v aplikační poznámce AP0025 – Komunikace v síti ARION – definice tabulkou.

2. Protokol ARION

ARION je komunikační protokol vyvinutý pro komunikaci řídicích systémů firmy AMiT s rozšiřujícími moduly. Pomocí těchto modulů lze navýšit počet vstupů/výstupů (jak číslicových tak analogových) řídicího systému. Maximální počet rozšiřujících modulů připojených do jedné komunikační sítě je 63, čímž lze dosáhnout rozšíření řídicího systému až o 1512 číslicových (63 × **DM-DI24**) nebo 756 analogových vstupů (63 × **DM-AI12**). V kapitole 9 je uveden přehled dostupných produktů firmy AMiT, jejich označení, typů a možností.

ARION je otevřený protokol, který podporují i jiní výrobci. Použití jiných zařízení než těch, které vyrábí firma AMiT, doporučujeme konzultovat s technickou podporou firmy AMiT. Přehled kompatibilních zařízení od jiných výrobců je uveden v kapitole 10.

Protokol ARION je sériový poloduplexní protokol a z toho vyplývají jistá omezení. Při vzrůstajícím počtu připojených zařízení narůstají časové nároky na přenos dat z a do modulů a tomu odpovídá i prodlužující se perioda možné komunikace s připojenými moduly. Výpočet minimální periody, se kterou je možné komunikovat s připojenými moduly, je uveden v kapitole 5.2.

2.1. Komunikační režimy protokolu ARION

Komunikace prostřednictvím protokolu ARION může probíhat čtyřmi různými režimy:

- Simplex
- Half-Duplex
- Duplex
- Autonomous

Pro komunikaci s rozšiřujícími moduly firmy AMiT se používá výhradně režim Half-Duplex nebo Autonomous.

Režim Half-Duplex je použitelný na komunikačním rozhraní RS485 s maximálním počtem 63 připojených modulů. Používá se obousměrný poloduplexní přenos. Master vysílá rámce slaveům a selektivně si na některé rámce může vyžádat odpověď.

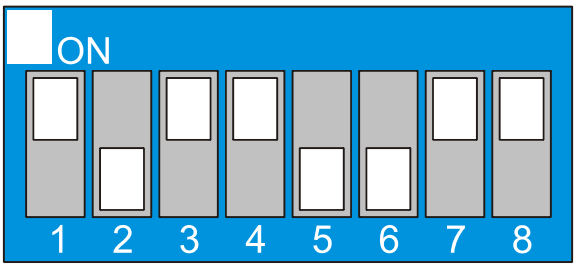
Popis ostatních režimů, včetně dalších informací, lze nalézt v související dokumentaci uvedené v úvodu této aplikační poznámky.

Poznámka
Komunikační rozhraní řídicího systému, kam je připojena síť ARION, již nelze použít pro připojení zařízení s jiným protokolem.

3. HW rozšiřujících modulů

3.1. HW konfigurace rozšiřujících modulů

Modulům **DM-xxx** je nutné nastavit adresu (musí být jedinečná v rámci sítě ARION) a komunikační rychlost, která musí být shodná s komunikační rychlostí zadanou v řídicím systému SW modulem ARION.



Obr. 1 - Nastavení komunikačních parametrů rozšiřujících modulů

Nastavení těchto parametrů se provádí pomocí přepínačů umístěných na rozšiřujících modulech.

Přepínače 1 .. 6 slouží k nastavení adresy modulu v síti ARION. Adresa může nabývat hodnot 1 .. 63. Adresa 0 není povolena (je vyhrazena pro MASTER).

Poznámka
*Některé rozšiřující moduly mohou zabírat více adres (např. **DM-PDO6NI6**).*

Pro nastavení komunikační rychlosti slouží přepínače 7 a 8 (7 = BAUD0, 8 = BAUD1). Váhy jednotlivých přepínačů adresy i možnosti nastavení komunikační rychlosti jsou uvedeny v následující tabulce.

Váhy přepínačů

ADR0	Váha 1
ADR1	Váha 2
ADR2	Váha 4
ADR3	Váha 8
ADR4	Váha 16
ADR5	Váha 32

BAUD0	**BAUD1**	**Rychlost komunikace**
OFF	OFF	9600 Bd
ON	OFF	19200 Bd
OFF	ON	38400 Bd
ON	ON	57600 Bd

Na výše uvedeném obrázku je tedy nastavena adresa modulu 13 a komunikační rychlost 57600 Bd.

Poznámka
Veškeré změny polohy jednotlivých přepínačů se projeví až po restartu rozšiřujícího modulu (odpojení a připojení napájení).

3.2. Význam LED

Všechny rozšiřující moduly jsou vybaveny indikačními LED, které umožňují vizuální kontrolu činnosti. V následujících tabulkách jsou uvedeny popisy jejich funkcí pro různé moduly.

Systémové LED

Modul	LED	Funkce
Všechny moduly	PWR	Svítí v případě připojeného napájecího napětí.
	RUN	Bliká s periodou cca 2 s (1:1) – indikuje funkční stav modulu.

Komunikační LED

Modul	LED	Funkce
Všechny moduly	RxD	Svítí při přijmu dat ze sítě ARION.
	TxD	Svítí při vysílání dat do sítě ARION.

LED indikující stav vstupů / výstupů

Modul	LED	Funkce
DM–DI24	DI0 .. DI23	Svítí v případě, že na příslušný vstup je přivedena hodnota log. 1.
DM–DO18	DO0 .. DO17	Svítí v případě, že na příslušném výstupu je nastavena hodnota log. 1.
DM–RDO12	RL0 .. RL11	Svítí v případě, že příslušný reléový výstup je sepnut.
DM–AI12	AI0 .. AI11	Svítí v případě, že hodnota přivedená na vstup je v rozmezí [min .. max] rozsahu AD převodníku.(1)
DM–AO8U(I)	AO0 .. AO7	Svítí v případě, že hodnota přivedená na výstup je větší než cca 0V (0mA) nebo podle zvoleného nastavení.(2)
DM-PDO6NI6	NI0 .. NI5 DO0 .. DO5	Svítí v případě, že je připojeno čidlo Ni1000 (vstupy NIx) nebo když je hodnota na výstupu v log. 1 (výstupy DOx).

Podle chování indikačních LED se dá v některých případech vizuálně vyhodnotit příčina vzniklého problému.

Poznámka

(1) *Hodnota je (30) až (max. - 30) z rozsahu převodníku.*

(2) *Chování LED modulu lze programově nastavit. Toto se děje zápisem na „fiktivní“ analogové výstupy AO.8 (význam MEZ1) a AO.9 (význam MEZ2). Chování LED je potom dané následujícím algoritmem:*
(MEZ2 > MEZ1) a současně (AOx >= MEZ2) => LED bliká
jinak je-li
(MEZ2 >= MEZ1) a současně (AOx > MEZ1) => LED svítí
jinak => LED nesvítí.
Pokud se do výstupů AO8 a AO9 nezapíšou žádné hodnoty (tj. budou nulové), budou dané LED svítit pro hodnoty AOx větší než nula.
Například pro analogové výstupy 0 .. 10 V bude nastavena MEZ1 = 4 V a MEZ2 = 8 V (tyto hodnoty se přenesou do daného rozšiřujícího modulu). Zápisem hodnoty 0 .. 4 V (včetně) do AOx (AO0 .. AO7) bude daná LED zhasnutá. Pro hodnotu v rozmezí 4 .. 8 V bude LED svítit a pro hodnoty nad 8 V (včetně) bude LED blikat.

4. Realizace sítě ARION

Pro správné plnění požadované funkce celé sítě ARION je nutno správně navrhnout, zapojit, nakonfigurovat jednotlivé moduly sítě a naprogramovat komunikaci.

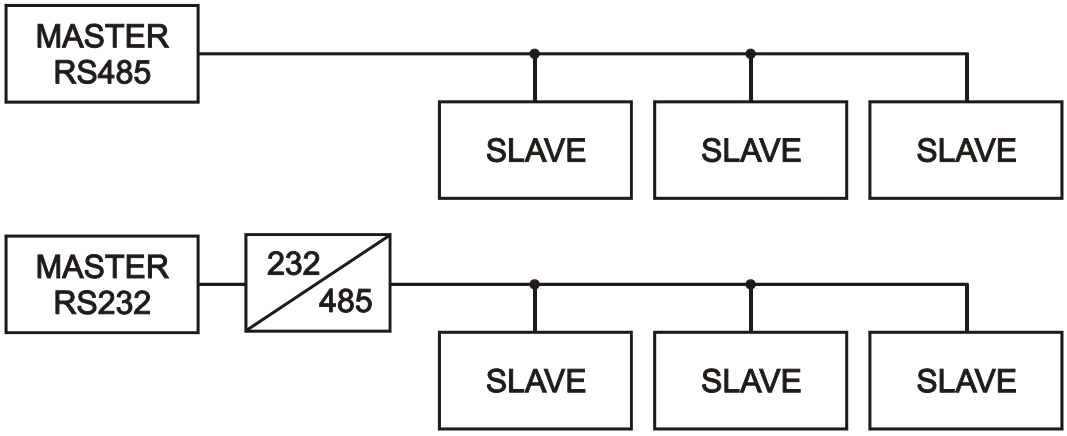
Při zapojování sítě sériovou linkou RS485 je nutno dodržet několik omezení a dbát na kvalitní provedení kabeláže, která má zásadní vliv na správnou funkci. Kromě kabeláže má vliv na funkci sítě i kvalitní napájení prvků. Je doporučeno použití stabilizovaných zdrojů a v zarušeném prostředí použití přepěťových ochran.

Zapojení komunikační sítě
Každé připojené zařízení musí mít nastavenou adresu a komunikační rychlost. Adresa musí být jedinečná pro každý rozšiřující modul na síti. Komunikační rychlost musí mít všechny rozšiřující moduly shodnou. Adresa i komunikační rychlost se nastavuje přepínačem na každém rozšiřujícím modulu.

Softwarová konfigurace
Po zapojení sítě je nutné provést programovou konfiguraci řídicího systému **(viz kapitola 6).**

4.1. Zapojení komunikační sítě

Komunikace prostřednictvím protokolu ARION probíhá po lince RS485 a je typu MASTER – SLAVE (Multi SLAVE). K řídicímu systému lze rozšiřující moduly připojit přímo na rozhraní RS485 nebo na rozhraní RS232 přes převodník (např. **DM-232TO485**). Obě tyto možnosti ukazuje následující obrázek.



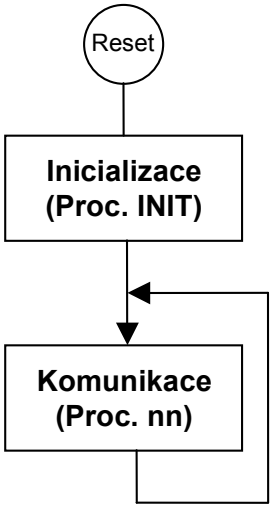
Obr. 2 - Připojení rozšiřujících modulů k řídicímu systému

Poznámka
*Převodník **DM-232TO485** připojený k RS232 řídicího systému AMiT se nastavuje jako řízený signálem RTS a při připojení převodníku k RS232 počítače se nastavuje jako automaticky řízený.*

Potřebné informace o lince RS485 naleznete v dokumentaci AP0016 – Zásady používání RS485.

5. Časové poměry

Při startu programu, který má komunikovat s rozšiřujícími moduly prostřednictvím protokolu ARION, je nutné celou síť inicializovat a nakonfigurovat. Při inicializaci řídicí systém posílá do jednotlivých modulů na síti rámce dat, které obsahují informace o režimech komunikace, a čeká na jejich odpověď. Pokud odpověď nedostane do uplynutí stanovené doby, prohlásí, že modul není připojen a pokračuje dalším modulem. Celou inicializaci je nutné provést programově v procesu INIT (popsáno dále). Po ní se již může odehrávat vlastní komunikace s moduly, která probíhá obvykle v periodickém procesu tak, jak ukazuje následující diagram.



Obr. 3 - Posloupnost činností v síti ARION po restartu řídicího systému

5.1. Doba inicializace sítě

5.1.1 Doba inicializace úplné sítě

Doba inicializace sítě je čas, za který řídicí systém rozešle všem modulům komunikační parametry a obdrží od všech modulů jejich potvrzení. Úplnou sítí se rozumí taková síť, kde všechny moduly, které se v programu inicializují, jsou fyzicky na síti přítomny a správně fungují (je správně připojen kabel RS485, je zapnuté napájení a modulům je správně nastavena adresa a komunikační rychlost).

Doba nutná k inicializaci sítě je přímo úměrná počtu inicializovaných modulů a nepřímo úměrná komunikační rychlosti. Následující tabulka udává dobu nutnou k inicializaci sítě.

Doba inicializace úplné sítě ARION

Doba inicializace sítě [ms]	
9600 Bd	T = 30 × N + 11
19200 Bd	T = 17 × N + 4
38400 Bd	T = 10 × N + 3
57600 Bd	T = 7 × N + 5

Kde N je počet modulů, které jsou inicializovány.

5.1.2 Doba inicializace neúplné sítě

Pokud se stane, že některé moduly, které jsou v programu inicializovány, nejsou na síti fyzicky přítomny, není připojen síťový kabel, napájení nebo je špatně zvolena adresa nebo komunikační rychlost – doba inicializace sítě se prodlouží o určitý čas za každý nezapojený modul tak, jak je uvedeno v následující tabulce:

Doba, o kterou se prodlouží inicializace při jednom chybějícím modulu [ms]	
9600 Bd	10
19200 Bd	15
38400 Bd	18
57600 Bd	20

Příklad
Při komunikační rychlosti 19200 Bd a sedmi připojených rozšiřujících modulech bude doba inicializace sítě trvat cca 120 ms. Pokud u tří modulů nebude zapojeno napájení, inicializace se prodlouží o 45 ms. Teoreticky je sice možné vyslat rámec dat do rozšiřujícího modulu (na jeho adresu), ale nelze již čekat, že bude odpovídat.

Poznámka
Není možné se v programu dotazovat na neinicializovaný modul!

5.2. Výpočet minimální periody komunikace s moduly

Doba, která uplyne od vložení požadavku na komunikaci do fronty (tedy od vyvolání příslušné funkce) po úspěšné získání nebo vložení dat, je závislá především na typu modulu (moduly zpracovávající analogové signály mají delší dobu vybavení než moduly zpracovávající signály číslicové), dále na momentálním vytížení komunikační linky (ve frontě mohou ještě čekat na vyřízení předchozí požadavky) a v neposlední řadě na rychlosti komunikace (s narůstající rychlostí komunikace se tato doba snižuje).

Pro výpočet minimální periody lze použít zjednodušený vzorec z následující tabulky.

Výpočet minimální periody komunikace s moduly

Přenosová rychlost	Minimální perioda komunikace [ms]
9600 Bd	T = 50 × DIG + 150 × ANL
19200 Bd	T = 25 × DIG + 80 × ANL
38400 Bd	T = 15 × DIG + 45 × ANL
57600 Bd	T = 12 × DIG + 35 × ANL

Kde DIG je počet modulů zpracovávajících číslicový signál a ANL je počet modulů zpracovávajících analogový signál.

Takto vypočítaný údaj je minimální možná perioda procesu, ve kterém se obsluhuje síť připojených modulů. Při zkrácení této periody nelze zaručit správnou funkci připojených V/V modulů.

Poznámka
Výpočet se provádí pro všechny definované moduly ARN_NODE uvedené v procesu Init a vypočtená minimální doba periody dokumentace MUSÍ být dodržena ve všech procesech, v nichž se vyskytují komunikační moduly ARN_xxx. Typickým příkladem, kdy toto není dodrženo, je rozdělení komunikace digitálních a analogových modulů do dvou procesů. Perioda komunikace digitálních modulů je krátká a často může klesnout pod periodu komunikace všech modulů. Při komunikaci pak dochází ke ztrátě požadavků a nekorektní činnosti.

Příklad
Mějme síť složenou z pěti rozšiřujících modulů zpracovávajících číslicový signál a dvou rozšiřujících modulů zpracovávajících analogový signál. Pro zvolenou komunikační rychlost 19200 Bd je minimální perioda komunikace

T = 25 × 5 + 80 × 2 = 285 ms.

Při pokusu o komunikaci s menší periodou nelze zaručit, že síť bude plně funkční. Pokud dosažený výsledek nevyhovuje požadavkům, není možno protokol ARION použít.

Dalším omezením při použití rozšiřujících modulů komunikujících protokolem ARION je nemožnost reagovat na přicházející impulzy, jejichž doba trvání je menší, než perioda komunikace (např. není stoprocentně zaručeno zachycení krátkého stisku tlačítka připojeného bez dalších tvarovacích obvodů na číslicový vstup modulu). Všechna tato omezení jsou podrobně popsána dále.

5.3. Doba první odezvy sítě

Je to časová prodleva od startu uživatelského programu v řídicím systému (a tedy začátku inicializace sítě) do proběhnutí první komunikace. Je to tedy minimální doba, za kterou je možno získat ze sítě platná data. Doba první odezvy je dána součtem doby inicializace sítě a periody komunikace s moduly.

5.4. Detekce ztráty spojení (Guardtime)

Hodnota parametru Guardtime udává čas, za jak dlouho po rozpadu komunikace výstupní moduly nastaví všechny výstupy do bezpečného stavu.

Pokud po definované době nedorazí do modulu žádný platný komunikační rámec, detekuje modul rozpad komunikace a v případě výstupního modulu se všechny jeho výstupy nastaví do bezpečného stavu. Bezpečný stav je dán napevno a nelze jej uživatelsky měnit.

Bezpečný stav pro různé typy výstupů

Typ výstupů	Bezpečný stav
Digitální výstupy	0 V
Reléové výstupy	rozepnuto
Analogové výstupy	0 V

Při pokusu o zapsání bezpečného stavu pomocí příkazu ARN_sfDO nebo ARN_sfAO do rozšiřujícího modulu **DM-xxx** (**NOA2x**), se tento modul chová jako na standardní příkaz ARN_DO, případně ARN_AO.

V případě rozpadu komunikace a nastavení výstupů na bezpečné hodnoty se po obnovení komunikace opět nastaví požadované hodnoty, nejdříve však za dobu rovnou periodě komunikace s moduly.

Periodu detekce ztráty spojení je vždy nutno nastavovat s ohledem na vytížení komunikační sítě a požadavky řízené technologie.

Při periodě komunikace s moduly kratší než 1 s by hodnota Guardtime měla být alespoň dvojnásobkem periody. U periody komunikace s moduly 10 s a delší by hodnota Guardtime měla být nastavena tak, aby řízená technologie byla v případě rozpadu komunikace včas zabezpečena vypnutím výstupů.

Poznámka
Pro zakázání detekce ztráty spojení se nastavuje hodnota 0.

6. SW konfigurace

Nastavení správné periody obsluhy jednotlivých rozšiřujících modulů má zásadní vliv na správnou funkci celé sítě ARION. Výpočet periody je uveden v kapitole 5.2. Je doporučeno takto vypočítanou hodnotu použít jako minimální periodu procesu, ve kterém je komunikace obsluhována. V žádném případě by se takto vypočítaná perioda neměla zkracovat. Při zkrácení periody nelze zaručit správnou funkci celé sítě.

Správné nastavení Guardtime může podstatným způsobem ovlivnit správnou funkci celé sítě. Nastavení Guardtime by mělo vycházet především z požadavků technologie. Hodnota se nastaví podle toho, za jak dlouho je požadováno, aby se výstupy v případě problémů s komunikací nastavily na bezpečné hodnoty. Pokud je však perioda obsluhy modulů menší než 1 s, nesmí být hodnota Guardtime menší než dvojnásobek periody. U periody obsluhy větší než 10 s je nutno vycházet z požadavku technologie.

6.1. Programová obsluha

Pro komunikaci protokolem ARION se řídicí systém konfiguruje v návrhovém prostředí DetStudio. Pro správnou funkci je nutno použít DetStudio verze 1.0.45 nebo vyšší.

6.1.1 Inicializace sítě

Konfigurace (a samotné sestavení) komunikační sítě probíhá vždy na začátku programu, v procesu INIT. K tomu slouží modul ARION, pomocí kterého se definuje globální nastavení sítě a moduly ARN_NODE, které definují typy jednotlivých rozšiřujících modulů.

Vytvoříme komunikační síť, sestávající se ze sedmi zařízení SLAVE a jednoho MASTER. Konfiguruje proběhne v procesu INIT:

```
:10000 ARION 1, 19200, HalfDupl4
:10001 ARN_NODE :10000, 1, 1000, @StavDI1, 2, 24, 0x000C
:10002 ARN_NODE :10000, 2, 1000, @StavDI2, 2, 24, 0x000C
:10003 ARN_NODE :10000, 3, 1000, @StavDO1, 3, 18, 0x000C
:10004 ARN_NODE :10000, 4, 1000, @StavDO2, 3, 18, 0x000C
:10005 ARN_NODE :10000, 5, 1000, @StavRDO, 3, 12, 0x000C
:10006 ARN_NODE :10000, 6, 1000, @StavAI1, 0, 12, 0x000C
:10007 ARN_NODE :10000, 7, 1000, @StavAI2, 0, 12, 0x000C
```

Síť ARION je parametrizována pro obsluhu sedmi zařízení – dvou modulů **DM-DI24**, dvou modulů **DM-DO18**, jednoho modulu **DM-RDO12** a dvou modulů **DM-AI12** s komunikační rychlostí 19200 Bd a režimem Half-Duplex po lince RS485. Je zaručeno, že všechna zařízení detekují ztrátu spojení s řídicím systémem do 1000 milisekund. Aktuální stav spojení je ukládán do proměnných @Stavxxx. Moduly analogových vstupů mají nakonfigurován počet datových bitů, na které je v komunikačním rámci uložena analogová hodnota na 12 a také, že hodnota v komunikačním rámci se chápe jako kladné číslo bez znaménkového bitu (bipolární = NE).

V příkladu jsou označena návěští modulů ARN_NODE, která odkazují na příslušný modul ARION.

Poznámka
Vlastní komunikaci není možné zahájit dříve, než proběhne celá inicializace sítě!

6.2. Periodická komunikace s rozšiřujícími moduly

Vlastní komunikaci je možné zahájit až poté, co proběhne celá inicializace sítě. Podle způsobu, jakým je tato podmínka splněna, rozlišujeme komunikaci na kontrolovanou a nekontrolovanou.

Kontrolovaná komunikace (doporučeno)
Takováto komunikace je povolena až v okamžiku, kdy se stavový bit modulu ARN_NODE nastaví na hodnotu 1 a tím je potvrzeno, že spojení s rozšiřujícím modulem bylo skutečně úspěšně navázáno a modul je připraven ke komunikaci. Při větším počtu modulů je nutno testovat před začátkem komunikace stavové bity všech modulů ARN_NODE.

Testováním stavových bitů těchto funkcí v průběhu programu lze detekovat ztrátu spojení s rozšiřujícím modulem a tedy poruchu.

Nekontrolovaná komunikace
Stavové bity jednotlivých modulů ARN_NODE nejsou testovány a komunikace je zahájena od uplynutí určitého časového intervalu od začátku inicializace (je potřeba řešit správným nastavením ofsetu periodického procesu – viz nápověda k DetStudiu odkaz Teorie\Procesy.

U tohoto způsobu komunikace je nutno ovšem počítat při volbě velikosti ofsetu procesu také s možností, že některé moduly nebudou připojeny (doba inicializace se tím prodlouží). Pokud nejsou kontrolovány stavové bity modulu ARN_NODE, nelze detekovat ztrátu spojení (poruchu).

Poznámka
Pro celkovou přehlednost a lepší funkčnost programu je vhodné zapisovat (číst) do (ze) všech připojených rozšiřujících modulů v jednom periodickém procesu (s výjimkou modulů, jejichž perioda komunikace se výrazně liší).

6.3. Popis modulů pro komunikaci

Pro vlastní komunikaci má návrhové prostředí implementovány samostatné moduly dle vykonávané funkce. Ty se volají vždy, když je potřeba číst stav vstupů (číslicových nebo analogových), zapisovat na výstupy (číslicové nebo analogové) případně i jiné funkce (např. obsluha sítě 868 MHz – viz dokument AP0021 – Komunikace v síti 868 MHz). Podrobnější informace o názvech základních modulů dává následující tabulka:

Seznam modulů dle jejich funkce

Operace	Název modulu
Čtení číslicových vstupů	ARN_DI
Čtení analogových vstupů	ARN_AI, ARN_NumAI
Zápis číslicových výstupů	ARN_DO
Zápis analogových výstupů	ARN_AO, ARN_NumAO

Zatímco modul ARN_NODE je vhodné volat v procesu INIT, moduly pro čtení nebo zápis jsou zpravidla volány v periodickém procesu. Uskutečnění fyzického přenosu po síti dosáhneme dosazením hodnoty 1 za parametr Přenést. Parametr Počátek udává číslo referenčního výstupu – tomu při čtení/zápisu přísluší nultý bit příslušné proměnné. Při zápisu se stav výstupů s číslem nižším, než je referenční, nemění.

Poznámka
Podrobný popis modulů pro komunikaci v síti ARION naleznete v nápovědě k vývojovému prostředí DetStudio.

ARN_DI
Modul ARN_DI má jeden z výstupních parametrů jménem Proměnná. Po uskutečnění přenosu po síti je v této proměnné (nejlépe typu L – lze najednou přenést všechny signály z číslicových

vstupů) hodnota odpovídající stavům vstupů modulu. Do parametru Uzel je nutno zadat návěští příslušného uzlu ARN_NODE, se kterým se komunikuje. Parametr Stav, který se při vložení požadavku do fronty resetuje do 0, se nastaví do jedničky, jestliže přenos proběhl v pořádku.

```
ARN_DI :10001, 1, @PrenDI1, 24, 0, Čtení[0,0]
        │      │  │        │   │  └ Stav vstupů
        │      │  │        │   └ Číslo prvního signálu
        │      │  │        └ Počet čtených signálů
        │      │  └ Stavová proměnná
        │      └ Povolení fyzického přenosu po síti
        └ Návěští příslušného uzlu ARN_NODE je povinné
```

ARN_DO

Modul ARN_DO má jeden z vstupních parametrů jménem Proměnná. Po uskutečnění přenosu po síti se binární hodnota proměnné uvedeného parametru zapíše na příslušné výstupy modulu. Do parametru Uzel je nutno zadat návěští příslušného uzlu ARN_NODE, se kterým se komunikuje. Parametr Stav se nastaví do jedničky, jestliže přenos proběhl v pořádku. Maximální smysluplná hodnota zapisované proměnné je rovna 2n - 1, kde n je počet výstupů modulu, tedy pro 18 výstupů je to 262143.

```
ARN_DO :10003, 1, @PrenDO1, 18, 0, Zápis[0,0]
        │      │  │        │       └ Zapisovaná proměnná
        │      │  │        └ Počet zapisovaných signálů
        │      │  └ Stavová proměnná
        │      └ Povolení fyzického přenosu po síti
        └ Návěští příslušného uzlu ARN_NODE je povinné
```

ARN_AI, ARN_NumAI

Jestliže máme v úmyslu číst z modulu analogových vstupů pomocí modulu ARN_AI (ARN_NumAI) více vstupů než jeden, použijeme konstrukci z následujícího příkladu. Za parametr Přenést zadáme hodnotu log. 1 jen při čtení prvního signálu – tj. u signálu ze vstupu AI0. Jeho velikost se uloží do proměnné Signal0. Při čtení dalších signálů (AI1 .. AI11) již za parametr Přenést zadáme hodnotu log. 0 a postupujeme stejně. Při čtení pouze jednoho vstupu postupujeme obdobně. Všechny ostatní parametry jsou shodné jako u předchozích modulů. Významy parametrů Rozsah .. FyzMax jsou vysvětleny v nápovědě k vývojovému prostředí DetStudio v popisu příslušných modulů. Moduly ARN_AI a ARN_NumAI lze mezi sebou zaměňovat. Rozdíl v nich je pouze ten, že se do parametru Hodnota zapisuje přímo číselná hodnota vstupního signálu (ARN_NumAI) nebo hodnota přepočtená do fyzikálních jednotek (ARN_AI).

```
ARN_AI      :10005, 1, @PrenAI0, 0, Signal0[0,0], ...
             │      │  │        │  └ Proměnná pro uložení
             │      │  │        └ Ukládá se signál ze vstupu 0
             │      │  └ Stavová proměnná
             │      └ Provádí se fyzicky přenos po síti
             └ Návěští příslušného uzlu ARN_NODE je povinné

ARN_AI      :10005, 0, NONE.0, 1, Signal1[0,0], ...
ARN_NumAI   :10005, 0, NONE.0, 2, Signal2[0,0]
ARN_AI      :10005, 0, NONE.0, 3, Signal3[0,0], ...
                    │  │       │  └ Proměnné pro uložení
                    │  │       └ Ukládá se signál ze vstupu č. n
                    │  └ Stavová proměnná – není nutné zadávat
                    └ Neprovádí se fyzicky přenos po síti
```

ARN_AO, ARN_NumAO

Podobně postupujeme i při zápisu více analogových výstupů pomocí modulu ARN_AO (ARN_NumAO). Do parametru Přenést zadáme hodnotu log. 1 jen při zápisu posledního signálu. Ve všech předchozích voláních modulu bude tedy parametr Přenést roven nule. Modul ARN_AO

a ARN_NumAO lze také mezi sebou zaměňovat. Rozdíl v nich je pouze ten, že se do parametru Hodnota zadává přímo číselná hodnota výstupního signálu (ARN_NumAO) nebo hodnota ve fyzikálních jednotkách (ARN_AO).

```
ARN_AO      :10010, 0, NONE.0, 0, Signal0[0,0], ...
ARN_NumAO   :10010, 0, NONE.0, 1, Signal1[0,0]
ARN_AO      :10010, 0, NONE.0, 2, Signal2[0,0], ...
                    │  │       │  └ Proměnné pro uložení
                    │  │       └ Ukládá se signál ze vstupu č. n
                    │  └ Stavová proměnná – není nutné zadávat
                    └ Fyzický přenos po síti zakázán

ARN_AI      :10010, 1, @PrenAO3, 3, Signal3[0,0], ...
              │      │  │        │  └ Proměnná pro uložení
              │      │  │        └ Ukládá se signál ze vstupu 0
              │      │  └ Stavová proměnná
              │      └ Fyzický přenos po síti povolen
              └ Návěští příslušného uzlu ARN_NODE je povinné
```

Poznámka

Popis modulů potřebných pro komunikaci v síti 868 MHz naleznete v nápovědě k vývojovému prostředí DetStudio nebo v dokumentu AP0021 – Komunikace v síti 868 MHz.

7. Ukázkové aplikace

Součástí přílohy ap0005_cz_02.zip jsou následující aplikace, vytvořené pro řídicí systém **StartKit**, ve vývojovém prostředí DetStudio.

7.1. Příklad č. 1 – Periodická komunikace s V/V moduly

V/V moduly:

1 × **DM-DO18**	adresa 1
1 × **DM-DI24**	adresa 2
1 × **DM-RDO12**	adresa 3
1 × **DM-AO8U**	adresa 4
1 × **DM-AI12**	adresa 5
1 × **DM-PDO6Ni6**	adresa 6 a 7

Parametry komunikace:

Rychlost	38400 Bd
Port	1 (RS485)
Režim	3 (Half-Duplex RS485)

Funkce připojených modulů:

DM-DO18	zapisuje se na všechny výstupy
DM-DI24	načítají se všechny vstupy
DM-RDO12	zapisuje se na všechny výstupy
DM-AO8U	zapisuje se na výstupy 0, 1 a 5
DM-AI12	načítají se vstupy 0, 6 a 11
DM-PDO6Ni6	načítají se Ni vstup 0 a řídí se PDO výstup 0

Příklad ukazuje rozšíření V/V prostoru řídicího systému o 24 DI, 18 DO, 12 RDO, 8 AO(U), 12 AI, 6 PDO a 6 Ni1000. Komunikuje se periodicky v Normal procesu. Stav komunikace s rozšiřujícími moduly je sledován v 0. bitu proměnných:

StavDO18
StavDI24
StavRDO12
StavAO8U
StavAI12
StavPDO6Ni6

V/V moduly jsou v programu reprezentovány proměnnými:

DO18	L	Nastavení dig. výstupů 0 .. 17
DI24	L	Stav dig. vstupů 0 .. 23
RDO12	I	Spínání relé 0 .. 11
AO8U	MF[8,1]	Nastavení napětí výstupu 0 .. 7
AI12	MF[12,1]	Čtení napětí na vstupu 0 .. 11
PDO6	MF[6,1]	Nastavení střídy výstupu 0 .. 5 hodnoty 0 .. 100 => 0 .. 100 %
Perioda	F	Nastavení periody PWM hodnoty 0 .. 100 => 0 .. 100 s
Ni6	MF[6,1]	Čtení hodnot napětí ze vstupu 0 .. 5
NiT	MF[6,1]	Přepočtená teplota snímače Ni1000

Výpočet periody komunikace

Pro stanovení minimální periody, při zvolené rychlosti, použijeme následující vzorec (viz kapitola 5.2):

T = 15 × DIG + 45 × ANL

DIG = 3 (**DM-DO18**, **DM-DI24**, **DM-RDO12**)

ANL = 4 (**DM-AO8U**, **DM-AI12**, **DM-PDO6NI6** × 2)

T = 15 × 3 + 45 × 4 = 45 + 180 = 225 ms

Minimální perioda procesu, ve kterém lze komunikaci obsluhovat, je přibližně 250 ms.

Poznámka
*Modul **DM-PDO6NI6** je nutno brát v tomto případě jako 2 analogové moduly (neboť je definován jako typ AI a AO).*

Guardtime
Pro tento příklad byla použita hodnota Guardtime 10 s, nejkratší možný Guardtime dle výpočtu je 480 ms (tj. dvojnásobek minimální periody komunikace).

Vlastní program je:

Proces Init
V procesu Init je nadefinována komunikace ARION požadovaných parametrů a všechny použité rozšiřující moduly takto:

```
// Zakladni definice site Arion
:1000 ARION 1, 38400, 3

// Definice uzlu pro DM-DO18 na adrese 1
:1001 ARN_NODE :1000, 1, 10000, StavDO18.0, 3, 18, 0x000C
// Definice uzlu pro DM-DI24 na adrese 2
:1002 ARN_NODE :1000, 2, 10000, StavDI24.0, 2, 24, 0x000C
// Definice uzlu pro DM-RDO12 na adrese 3
:1003 ARN_NODE :1000, 3, 10000, StavRDO12.0, 3, 12, 0x000C
// Definice uzlu pro DM-AO8U na adrese 4
:1004 ARN_NODE :1000, 4, 10000, StavAO8U.0, 1, 8, 0x000C
// Definice uzlu pro DM-AI12 na adrese 5
:1005 ARN_NODE :1000, 5, 10000, StavAI12.0, 0, 12, 0x000C
// Definice PDO vystupu PDO6Ni6 na adrese 6 - PDO
:1006 ARN_NODE :1000, 6, 10000, StavPDO6Ni6.0, 1, 6, 0x000C
// Definice PDO vystupu PDO6Ni6 na adrese 7 - Ni1000
:1007 ARN_NODE :1000, 7, 10000, NONE.0, 0, 6, 0x000C
```

Proces Normal_0
Perioda procesu je 1 s. V tomto procesu se vyvolává vlastní komunikace s rozšiřujícími moduly **DM-xxx**.

```
// Zapis do DM-DO18
ARN_DO :1001, 1, @ZapisDO, 18, 0, DO18

// Cteni z DM-DI24
ARN_DI :1002, 1, @CteniDI, 24, 0, DI24

// Zapis do DM-RDO12
ARN_DO :1003, 1, @ZapisRDO, 12, 0, RDO12

// Zapis do DM-AO8U vystup 0, vystupni U rozsah 0-10V, prepocet 1:1
ARN_AO :1004, 0, NONE.0, 0, AO8U[0,0], 10.000, 0.000, 10.000, 0.000, 10.000
// Zapis do DM-AO8U vystup 1, vystupni U rozsah 0-10V, prepocet 10:1
ARN_AO :1004, 0, NONE.0, 1, AO8U[1,0], 10.000, 0.000, 10.000, 0.000, 100.000
// Zapis do DM-AO8U vystup 5, vystupni U rozsah 0-10V, prepocet 100:1
ARN_AO :1004, 1, @ZapisAO, 5, AO8U[5,0], 10.000, 0.000, 10.000, 0.000, 1000.000

// Cteni analogoveho vstupu 0 z DM-AI12 (0 .. 5 V), prepocet 1:1
ARN_AI :1005, 1, @CteniAI, 0, AI12[0,0], 5.000, 0.000, 5.000, 0.000, 5.000
// Cteni analogoveho vstupu 6 z DM-AI12 (0 .. 10 V), prepocet 1:1
ARN_AI :1005, 0, NONE.0, 6, AI12[6,0], 10.000, 0.000, 10.000, 0.000, 10.000
```

```
// Cteni analogoveho vstupu 11 z DM-AI12 (0 .. 10 V), prepocet 1:10
ARN_AI :1005, 0, NONE.0, 11, AI12[11,0], 10.000, 0.000, 10.000, 0.000, 100.000

// Zapis stridy do DM-PDO6NI6 vystupu 0
ARN_AO :1006, 0, NONE.0, 0, PDO6[0,0], 100.000, 0.000, 100.000, 0.000, 16384.000
// Zapis periody PWM modulu PDO6Ni6
ARN_AO :1006, 1, NONE.0, 6, Perioda, 10.000, 0.000, 10.000, 0.000, 16384.000

// Nacteni hodnoty teplomeru na vstupu Ni0 modulu PDO6Ni6
ARN_AI :1007, 1, NONE.0, 0, Ni6[0,0], 5.000, 0.000, 5.000, 0.000, 5.000
```

Proces Normal_1
Perioda procesu je 1 s. V tomto procesu se provádí převod hodnoty napětí ze vstupu Ni0 rozšiřujícího modulu **DM-PDO6NI6** na teplotu.

```
// Prepocet napeti prevodniku na teplotu
Ni1000U2T Ni6[0,0], NiT[0,0], 6180, 15.000, 3920.000
```

7.2. Příklad č. 2 – Různá perioda komunikace

V/V moduly

1 × **DM-DO18**	adresa 1
1 × **DM-DI24**	adresa 2
1 × **DM-RDO12**	adresa 3
1 × **DM-AO8U**	adresa 4
1 × **DM-AI12**	adresa 5

Parametry komunikace:

Kom. rychlost	38400 Bd
Kom. port	1 (RS485)
Režim	3 (Half-Duplex RS485)

Funkce připojených modulů:

DM-DI24	načítají se stavy čítačů na vstupech 7, 15 a 23 s možností nulování
DM-DO18	zapisuje se na všechny výstupy
DM-RDO12	zapisuje se na všechny výstupy
DM-AO8U	zapisuje se na výstupy 0, 1 a 5
DM-AI12	načítají se vstupy 0, 6 a 11

Poznámka
*U firmware **DM-DI24** verze 1.63 a starší je zrcadlově otočena každá sedmice čítačových vstupů DI (DI0 je na čísle kanálu 7, DI1 na kanálu 6, DI8 na kanálu 15, DI16 na kanálu 23 apod.). Od firmware verze 1.64 je toto již opraveno.*

Příklad ukazuje rozšíření V/V prostoru řídicího systému o 24 DI, 18 DO, 12 RDO, 8 AOU, 12 AI, 6 PDO a 6 Ni. Komunikuje se v rámci procesu Proc_Hi0 s periodou 250 ms, ale analogové moduly (8 AOU, 12 AI a čítače 24 DI) se komunikují každé padesáté spuštění procesu (tzn. s 12,5 s periodou). Bitem Logika.1 se povolí zápis hodnoty proměnné DI24Set do příslušných čítačů. Stav komunikace s moduly je sledován v 0. bitu následujících proměnných:

StavDO18
StavDI24
StavRDO12
StavAO8U
StavAI12

V/V moduly jsou v programu reprezentovány proměnnými:

DO18	L	Nastavení dig. výstupů 0 .. 17
DI24	MI[24,1]	Stav čítačů dig. vstupů 0 .. 23

RDO12	I	Spínání relé 0 .. 11
AO8U	MF[8,1]	Nastavení napětí výstupu 0 .. 7
AI12	MF[12,1]	Čtení napětí výstupu 0 .. 11
Cyklus	I	Počítáni počtu spuštění procesu
Logika.0	Bit	Pro rozhodování, zda se mají přenést analogová data
Logika.1	Bit	Pro rozhodování, zda se mají vynulovat čítače

Výpočet periody komunikace
Pro stanovení minimální periody použijeme následující vzorec (viz kapitola 5.2)
T = 15 × DIG + 45 × ANL

DIG = 2 (**DM-DO18**, **DM-RDO12**)
ANL = 3 (**DM-DI24**, **DM-AO8U**, **DM-AI12**)
T = 15 × 2 + 45 × 3 = 165 ms

Minimální perioda procesu, ve kterém lze komunikaci obsluhovat, je přibližně 180 ms.

Upozornění
I když se analogové moduly obsluhují pouze jednou za 12,5 s, do výpočtu bezpečné periody komunikace se musí zahrnout. Pokud tak neučiníme a nastavíme dobu periody komunikace kratší než doporučenou, může docházet ke ztrátě stavů digitálních signálů během komunikace analogových hodnot.

Guardtime
Pro tento příklad byla použita hodnota Guardtime 5 s (**DM-DO18**, **DM-RDO12**) a 30 s (**DM-DI24**, **DM-AO8U**, **DM-AI12**), nejkratší možný Guardtime je dle výpočtu přibližně 360 ms.

Proces Init
V procesu Init je nadefinována komunikace ARION požadovaných parametrů a všechny použité rozšiřující moduly takto:

```
// Definice komunikační sítě ARION
:1000 ARION 1, 38400, 3

// Definice uzlu pro DM-DO18 na adrese 1
:1001 ARN_NODE :1000, 1, 5000, StavDO18.0, 3, 18, 0x000C
// Definice uzlu pro čtení čítačů DM-DI24 na adrese 2
:1002 ARN_NODE :1000, 2, 30000, StavDI24.0, 0, 24, 0x000E
:1006 ARN_NODE :1000, 2, 0, NONE.0, 1, 24, 0x000E
// Definice uzlu pro DM-RDO12 na adrese 3
:1003 ARN_NODE :1000, 3, 5000, StavRDO12.0, 3, 12, 0x000C
// Definice uzlu pro DM-AO8U na adrese 4
:1004 ARN_NODE :1000, 4, 30000, StavAO8U.0, 1, 8, 0x000C
// Definice uzlu pro DM-AI12 na adrese 5
:1005 ARN_NODE :1000, 5, 30000, StavAI12.0, 0, 12, 0x000C
```

Proces Proc_Hi0
Perioda procesu je 250 ms. V tomto procesu se vyvolává vlastní komunikace s rozšiřujícími moduly **DM-xxx**. S moduly **DM-DO18** a **DM-RDO12** probíhá kontrolovaná komunikace (tj. probíhá pouze v případě správně inicializovaného rozšiřujícího modulu). Zkrácený výpis procesu:

```
//Kontrolovaná komunikace - zápis do DM-DO18
If StavDO18.0, :NONE
      ARN_DO :1001, 1, @ZapisDO, 18, 0, DO18
EndIf

//Čtení čítače na DI7 DM-DI24
ARN_AI :1002,1,@CteniDI,7,DI24[7,0],16384.000,0.000,16383.000,0.000,16383.000
...
// Zjištění, zda uběhla doba 12,5 s po které se mají přenést AO a AI
Let Logika.0 = IF(Cyklus<100, false, true)
If Logika.0, :NONE
```

```
      // Zápis do DM-AO8U výstup 0, výstupní U rozsah 0 .. 10 V, přepočet 1:1
      ARN_AO :1004, 0, NONE.0, 0,AO8U[0,0], 10.000, 0.000, 10.000, 0.000, 10.000
      ...
      // Čtení analogového vstupu 0 z DM-AI12 (0 .. 5 V), přepočet 1:1
      ARN_AI :1005, 1, @CteniAI, 0, AI12[0,0], 5.000, 0.000, 5.000, 0.000, 5.000
      ...
EndIf

Let Cyklus = If(Cyklus==100, 0, cyklus+1)

//Povolení vynulování hodnot čítačů 7,15 a 23
If Logika.1, :NONE
      // Zápis hodnoty DI24Stav[0,7] do čítače na DI7
      ARN_AO :1006,0,NONE.0,7,DI24Set[7,0],16384.0,0.0,16383.0,0.0,16383.0
      ...
      // Zajištění čítání v dalším vyvolání procesu
      Let Logika.1 = false
EndIf
```

7.3. Příklad č. 3 – Komunikace s ovladačem NOA21

Definice uzlů pro daný ovladač **NOA21** je:

Proces INIT

```
:1000 ARION 1, 19200, 3
:1001 ARN_NODE :1000, 1, 5000, NOA21_stav.0, 0, 24, 0x000C //AI
:1002 ARN_NODE :1000, 1, 0, NONE.0, 2, 8, 0x000C     //DI
:1003 ARN_NODE :1000, 1, 0, NONE.0, 3, 8, 0x000C     //DO
:1004 ARN_NODE :1000, 1, 0, NONE.0, 1, 24, 0x000C    //AO
```

Následující výpis programu ukazuje příklad komunikace s ovladačem **NOA21**.

Periodický proces

```
// Vyčtení teploty
ARN_AI :1001, 1, NOA21_stav.1, 0, Teplota[0,0], 10.0, 0.0, 10.0, -55.0, 125.0

// Vyčtení korekce v rozsahu -100% až 100%
ARN_AI :1001, 0, NONE.0, 1, Korekce[0,0], 10.0, 0.0, 10.0, -100.0, 100.0

// Načtení stavu (DI)
ARN_DI :1002, 1, NONE.0, 8, 0, Stavy

// Pokud došlo ke změně v NOA21 (Stavy.0 = true), akceptujeme to
If Stavy.0, :NONE
      //Pokud je NOA21 po resetu (Bit0..Bit6=true), nastavíme poslední stav z ŘS
      //jinak jen potvrdíme hodnotu z prom. Stavy
      Let Stavy = If ((Stavy & 0x7F) == 0x7F, Stavy_DO | 0x01, Stavy)
      // Uložení stavu pro displej
      Let Stavy_DO = Stavy & ~0x01
      // Nastavení příznaku komunikace pro zápis DO do NOA21
      Let NOA21_stav.3 = true
Else :NONE
      // Pokud došlo ke změně hodnoty v proměnné Stavy_DO, zapiš do NOA21
      VarWStat Stavy_DO, NOA21_stav.3, 0
      Let Stavy = If (NOA21_stav.3, Stavy_DO, Stavy)
EndIf

// Zápisu stavu (DO)
ARN_DO :1003, NOA21_stav.3, NONE.0, 8, 0, Stavy

// Volání zápisu jasu LED (při SetLED = true se zapíše hodnota JasLED)
ARN_AO :1004, SetLED.0, NONE.0, 2, JasLED, 10.0, 0.0, 10.0, 0.0, 4.0
```

```
If SetLED.0, :NONE
      Let SetLED = 0
EndIf
```

7.4. Příklad č. 4 – Komunikace s DM-UI8DO8 a DM-UI8RDO8

V/V moduly:

1 × **DM-UI8DO8** adresa 1
1 × **DM-UI8RDO8** adresa 4

Parametry komunikace:

Rychlost	19200 Bd
Port	1 (RS485)
Režim	3 (Half-Duplex RS485)

Funkce připojených modulů:

DM-UI8DO8 zapisuje se na všechny výstupy DO, čtou se všechny vstupy
DM-UI8RDO8 zapisuje se na všechny výstupy RDO, čtou se všechny vstupy

Nastavení universálních vstupů obou modulů:

UI0.0	0 .. 5 V
UI0.1	0 .. 10 V
UI0.2	0 .. 20 mA
UI0.3	Ni1000
UI0.4 .. UI0.7	DI

Příklad ukazuje rozšíření V/V prostoru řídicího systému o 8 DO, 8 RDO, 8 DI, 6 AI a 2 Ni1000. Komunikuje se periodicky v procesu typu Normal. Stav komunikace s rozšiřujícími moduly je sledován v 0. bitu proměnných:

UI8DO8_stav	pro modul **DM-UI8DO8**
UI8RDO8_stav	pro modul **DM-UI8RDO8**

Modul DM-UI8DO8 je v programu reprezentován proměnnými:

Data_UI	MF[4,1]	hodnoty z universálních výstupů UI0.0 .. UI0.3
Data_DI	I	hodnoty z universálních výstupů UI0.4 .. UI0.7
Data_DO	I	hodnoty zapisované na DO
Teplota	F	hodnota UI0.3 přepočítaná na teplotu

Modul DM-UI8RDO8 je v programu reprezentován proměnnými:

Data_UI2	MF[4,1]	hodnoty z universálních výstupů UI0.0 .. UI0.3
Data_DI2	I	hodnoty z universálních výstupů UI0.4 .. UI0.7
Data_DO2	I	hodnoty zapisované na RDO
Teplota2	F	hodnota UI0.3 přepočítaná na teplotu

Výpočet periody komunikace

Pro stanovení minimální periody, při zvolené rychlosti, použijeme následující vzorec (viz kapitola 5.2):

T = 25 × DIG + 80 × ANL

DIG = 4 (**DM-UI8DO8** × 2, **DM-UI8RDO8** × 2)

ANL = 2 (**DM-UI8DO8**, **DM-UI8RDO8**)

T = 25 × 4 + 80 × 2 = 100 + 160 = 260 ms

Minimální perioda procesu, ve kterém lze komunikaci obsluhovat, je přibližně 260 ms.

Proces Init

V procesu Init je nadefinována komunikace ARION požadovaných parametrů a všechny použité rozšiřující moduly takto:

```
//--- Inicializace komunikace ARION ---
:1000 ARION 1, 19200, 3

// --- Inicializace DM-UI8DO8 adresa 1 ---
:1001 ARN_NODE :1000, 1, 3000, UI8DO8_stav.0, 0, 8, 0x000C // AI
:1002 ARN_NODE :1000, 1, 0, NONE.0, 2, 8, 0x000C           // DI
:1003 ARN_NODE :1000, 1, 0, NONE.0, 3, 8, 0x000C           // DO

// --- Inicializace DM-UI8RDO8 adresa 4 ---
:1041 ARN_NODE :1000, 4, 3000, UI8RDO8_stav.0, 0, 8, 0x000C// AI
:1042 ARN_NODE :1000, 4, 0, NONE.0, 2, 8, 0x000C           // DI
:1043 ARN_NODE :1000, 4, 0, NONE.0, 3, 8, 0x000C           // DO
```

Periodický proces

```
// --- DM-UI8DO8 ---
// UI0.0 = typu AI - 0 .. 5 V
ARN_AI :1001, 1, NONE.0, 0, Data_AI[0,0], 5.000, 0.000, 5.000, 0.000, 100.000
// UI0.1 = typu AI - 0 .. 10 V
ARN_AI :1001, 0, NONE.0, 1, Data_AI[1,0], 10.000, 0.000, 10.000, 0.000, 100.000
// UI0.2 = typu AI - 0 .. 20 mA
ARN_AI :1001, 0, NONE.0, 2, Data_AI[2,0], 20.000, 0.000, 20.000, 0.000, 100.000
// UI0.3 = typu Ni1000
ARN_AI :1001, 0, NONE.0, 3, Data_AI[3,0], 5.000, 0.000, 5.000, 0.000, 5.000
//přepočet na teplotu
Ni1000U2T Data_AI[3,0], Teplota, 6180, 15.000, 3920.000
// UI0.4 .. UI0.7 = typu DI
ARN_DI :1002, 1, NONE.0, 4, 4, Data_DI

// Zápis DO
ARN_DO :1003, 1, NONE.0, 8, 0, Data_DO

// --- DM-UI8RDO8 ---
// UI0.0 = typu AI - 0 .. 5 V
ARN_AI :1041, 1, NONE.0, 0, Data_AI2[0,0], 5.000, 0.000, 5.000, 0.000, 100.000
// UI0.1 = typu AI - 0 .. 10 V
ARN_AI :1041, 0, NONE.0, 1, Data_AI2[1,0], 10.000, 0.000, 10.000, 0.000, 100.000
// UI0.2 = typu AI - 0 .. 20 mA
ARN_AI :1041, 0, NONE.0, 2, Data_AI2[2,0], 20.000, 0.000, 20.000, 0.000, 100.000
// UI0.3 = typu Ni1000
ARN_AI :1041, 0, NONE.0, 3, Data_AI2[3,0], 5.000, 0.000, 5.000, 0.000, 5.000
//přepočet na teplotu
Ni1000U2T Data_AI[3,0], Teplota2, 6180, 15.000, 3920.000
// UI0.4 .. UI0.7 = typu DI
ARN_DI :1042, 1, NONE.0, 4, 4, Data_DI2

// Zápis RDO
ARN_DO :1043, 1, NONE.0, 8, 0, Data_DO2
```

8. Nejčastější problémy

8.1. Nedaří se navázat komunikaci

Jsou všechny moduly připojeny na napájení?
Pokud ano, svítí na všech modulech LED PWR.

Jsou všechny moduly plně funkční?
Pokud ano, bliká na všech modulech LED RUN s periodou cca 2 s (1:1).

Je správně nastavena komunikační rychlost a adresa každého modulu?
Pokud ano, svítí nebo blikají (s různou periodou i střídou) LED RxD i TxD na příslušných modulech. Pokud bliká pouze LED RxD, znamená to, že síť je sice správně zapojená, ale je špatně nastavena komunikační rychlost nebo adresa daného modulu.

8.2. Komunikace je navázána, ale nefunguje spolehlivě

Je možné, že je přetížená komunikační síť.
Pokud ano, LED TxD na řídicím systému neustále svítí nebo zhasíná jen na krátkou dobu.

- Je možné, že komunikační síť zahltily požadavky na detekci ztráty spojení (parametr Kontrola v modulu ARN_NODE) – je nutno prodloužit jejich periodu.
- Při malé periodě komunikace s připojenými moduly, nízké komunikační rychlosti a větším počtu modulů může tato situace nastat – je nutno zvětšit periodu komunikace nebo zvýšit komunikační rychlost.

V okolí komunikačních linek se může vyskytovat silné rušení.

Nutno provést kabeláž linky RS485 v souladu s doporučením z dokumentu AP0016 – Zásady používání RS485 a je nutno použít vhodné přepěťové ochrany.

9. Dodatek A – Produkty firmy AMiT

9.1. Dostupné rozšiřující moduly firmy AMiT pro ARION

Dostupné rozšiřující moduly a ovladače pro ARION

Označení modulu	Typ modulu	Počet V/V
DM-DI24	Číslicové vstupy	24
DM-DO18	Číslicový výstupy	18
DM-RDO12	Reléové výstupy	12
DM-AI12	Analogové vstupy	12
DM-AO8I	Analogové výstupy	8
DM-AO8U	Analogové výstupy	8
DM-PDO6Ni6	Kombinovaný modul	6 × Ni1000 6 × PDO
DM-UI8DO8	Kombinovaný modul	8 × AI / DI / Ni1000 8 × DO
DM-UI8RDO8	Kombinovaný modul	8 × AI / DI / Ni1000 8 × RDO
NOAxx	Nástěnný ovladač	-
DM-RFC	Rx/Tx sítě 868 MHz	-
DM-OT	Převodník OpenTherm	-
DM-MPBUS	Převodník MP-Bus	-

Poznámka
Tento seznam modulů je aktuální k datu vytvoření této aplikační poznámky. Aktuální přehled všech dostupných modulů naleznete na www.amit.cz.
*Bližší informace k **DM-RFC** naleznete v AP0021 – Komunikace v síti 868 MHz.*
*Bližší informace k **DM-OT** naleznete v AP0028 – Zařízení OpenTherm v síti ARION.*
*Bližší informace k **DM-MPBUS** naleznete v AP0002 – Komunikace v síti MP-BUS.*

9.2. Využití funkce čítačových vstupů modulů DM-DI24

Od verze firmware modulů **DM-DI24** 1.63 je k dispozici funkce čítání příchozích impulzů na kterémkoli ze vstupů modulu. Tato alespoň částečně řeší problémy se zachycováním krátkých impulzů, např. z některých snímačů či ovládacích prvků. Při použití této funkce je ovšem nutno brát v potaz následující omezení:

- Vnitřní čítač impulzů je 32 b, hodnota přenášená protokolem ARION je ovšem omezena na 16383 (14 b).
- Maximální možná frekvence příchozích impulzů může být 25 Hz. Při frekvenci větší není možné zaručit, že všechny příchozí impulzy budou zaznamenány.
- Interní čítač modulu je nulován při odpojení napájecího napětí, případně je možno jej nulovat programově.
- Programově je nutno ošetřit přetečení interního čítače (tedy především přetečení přečtené hodnoty - prvních 14 bitů).

Parametrizace komunikace – Proces INIT:

```
:10000 ARION 1, 38400, HalfDupl4
:10001 ARN_NODE :10000, 1, 2000, StavDI24.0, 2, 24, 0x000C
:10002 ARN_NODE :10000, 1, 2000, NONE.0, 0, 24, 0x000E
:10003 ARN_NODE :10000, 1, 2000, NONE.0, 1, 24, 0x000E
                        └ Všechny uzly ARN_NODE mají stejnou adresu!
```

Všechny uzly ARN_NODE odkazují na stejnou adresu. Modulem ARN_DI (který se odkáže na návěští 10001) je možno kdykoli číst aktuální stav vstupů DI rozšiřujícího modulu, modulem ARN_AI se čte stav čítačů vstupů DI.

Proměnné Hodnota1 .. Hodnota3 zde musí být celočíselného typu na rozdíl od komunikace s moduly analogových vstupů.

Čtení hodnot vnitřních čítačů – Periodický proces:

```
ARN_AI :10002,1, @STAV,0,Hodnota1[0,0],16384.0,0V/0mA,16383.0,0.0,16383.0
ARN_AI :10002,0,NONE.0,1,Hodnota2[0,0],16384.0,0V/0mA,16383.0,0.0,16383.0
ARN_AI :10002,0,NONE.0,2,Hodnota3[0,0],16384.0,0V/0mA,16383.0,0.0,16383.0
                │ │       │ │              │        │       │        │   └ FyzMax
                │ │       │ │              │        │       │        └ FyzMin
                │ │       │ │              │        │       └ ElMax
                │ │       │ │              │        └ ElMin
                │ │       │ │              └ Rozsah
                │ │       │ └ Proměnná pro uložení hodnoty
                │ │       └ Pořadí přenášeného signálu
                │ └ Indikace stavu přenosu dat po sběrnici
                └ Pouze jeden fyzický přenos dat po sběrnici
```

Hodnoty parametrů Rozsah, ElMin, ElMax, FyzMin, FyzMax jsou zvoleny tak, aby výstupem funkce bylo celé číslo 0 .. 16383 udávající stav spodních 14 bitů vnitřního čítače každého vstupu.

Nastavení hodnot vnitřních čítačů – může být umístěno např. v podprogramu volaném při splnění vhodně nastavených podmínek (tj. při hrozícím přetečení interního čítače modulu).

```
ARN_AO :10003,1,@Nulovani,0,Nula[0,0],16384.0,0V/0mA,16383.0,0.0,16383.0
                │ │         │ │         │        │       │        │   └ FyzMax
                │ │         │ │         │        │       │        └ FyzMin
                │ │         │ │         │        │       └ ElMax
                │ │         │ │         │        └ ElMin
                │ │         │ │         └ Rozsah
                │ │         │ └ Proměnná pro uložení hodnoty
                │ │         └ Pořadí přenášeného signálu
                │ └ Indikace stavu přenosu dat po sběrnici
                └ Povolení fyzického přenosu dat po sběrnici
```

Tato konstrukce zabezpečí přenesení hodnoty proměnné Nula na čítač, který je připojen k vstupu DI0 modulu **DM-DI24**.

Poznámka
U firmware verze 1.63 a starší je u modulu DI24 zrcadlově otočena každá osmice čítačových vstupů DI (DI0 je na čísle kanálu 7, DI1 na kanálu 6, DI8 na kanálu 15, DI16 na kanálu 23 apod.). Od firmware 1.64 je toto již opraveno.

9.3. Použití modulu DM-PDO6Ni6

Jedná se o kombinovaný rozšiřující modul se šesti analogovými vstupy (přizpůsobenými pro připojení čidel Ni1000) a šesti číslicovými výstupy. Číslicové výstupy mohou být parametrizovány buď jako klasické číslicové výstupy DO a také jako PWM – PDO. U PWM výstupů lze měnit periodu modulace (v rozsahu 1 .. 100 s) a její střídu (v rozsahu 0 .. 100 %). Při startu modulu je přednastavena perioda na hodnotu 1 s a střída 0 %. Každý z číslicových výstupů je možno konfigurovat nezávisle do jednoho z uvedených režimů (DO nebo PDO).

Modul zabírá na síti 2 adresy (je tedy možno připojit pouze 31 takových modulů do jedné sítě ARION). Uzlu DO (PDO) se přiřazuje první z adres (nastavena na přepínačích modulu), uzlu Ni1000 adresa o 1 vyšší. Pokud je žádána obsluha PDO číslicových výstupů, je nutno nakonfigurovat uzel s první adresou na typ AO. Stavy spojení řídicího systému s modulem je

možno testovat pouze u uzlu s první adresou. V následujících příkladech je na přepínači modulu **DM-PDO6NI6** nastavena adresa 17. Jedná se vždy o výpis části kódu procesu INIT.

Příklad konfigurace všech číslicových výstupů jako DO:

```
:10000 ARION 1, 57600, HalfDupl4
:10001 ARN_NODE :10000, 17, 2000, Stav.0, 3, 6, 0x000C //DO (DO)
:10003 ARN_NODE :10000, 18, 2000, NONE.0, 0, 6, 0x000C //AI (Ni1000)
```

Příklad konfigurace všech číslicových výstupů jako PDO:

```
:10000 ARION 1, 57600, HalfDupl4
:10002 ARN_NODE :10000, 17, 2000, Stav.2, 1, 12, 0x000C //AO (PDO)
:10003 ARN_NODE :10000, 18, 2000, NONE.0, 0, 6, 0x000C  //AI (Ni1000)
```

Příklad konfigurace číslicových výstupů jako DO i jako PDO:

```
:10000 ARION 1, 57600, HalfDupl4
:10001 ARN_NODE :10000, 17, 2000, Stav.1, 3,  6, 0x000  //DO (DO)
:10002 ARN_NODE :10000, 17, 2000, NONE.0, 1, 12, 0x000C //AO (PDO)
:10003 ARN_NODE :10000, 18, 2000, NONE.0, 0,  6, 0x000C //AI (Ni1000)
```

Periodický proces

Jednotlivé číslicové výstupy jsou sice konfigurovatelné nezávisle na sobě, je ale vhodné výstupy stejného charakteru (DO nebo PDO) seskupit a používat vždy definici jen jednoho z typů. Ovládání šesti číslicových výstupů, které jsou konfigurovány např. jako: DO, PDO, DO, PDO, DO, PDO je zbytečně komplikované. Do výstupů PDO je vhodné zapisovat parametry pouze při jejich změně (kvůli menšímu zatížení výkonu řídicího systému a komunikační linky). V následujícím příkladu je ukázáno ovládání dvou skupin po třech výstupech.

Výstupy 0 .. 2 PDO

```
ARN_AO :10002, 0, NONE.0, 0, Strida[0,0], 16384.0, 0V/0mA, 16384.0, 0.0, 16384.0
ARN_AO :10002, 0, NONE.0, 1, Strida[1,0], 16384.0, 0V/0mA, 16384.0, 0.0, 16384.0
ARN_AO :10002, 0, NONE.0, 2, Strida[2,0], 16384.0, 0V/0mA, 16384.0, 0.0, 16384.0
```

Perioda modulace se nastavuje pro všechny PDO výstupy společně, a to zápisem na virtuální kanál 6. Může nabývat hodnot 0 .. 100 s a definice zápisu do tohoto kanálu může vypadat takto:

```
ARN_AO :10002, 1, Stav.0, 6, Perioda, 16384.0, 0V/0mA, 16384.0, 0.0, 16384.0
```

Výstupy 3 .. 5 DO

```
ARN_DO :10001, 1, Stav.4, 3, 3, Vystup[0,0]
```

Vstupy Ni1000

Pro čtení vstupů Ni1000 a jejich přepočet na teplotu ve °C můžeme použít následující příklad:

```
REM Čtení vstupů Ni1000
ARN_AI :10003, 1, Stav.5, 0, AI[0,0], 5V, 0V/0mA, 5V, 0.0, 5.0
ARN_AI :10003, 0, NONE.0, 1, AI[1,0], 5V, 0V/0mA, 5V, 0.0, 5.0
ARN_AI :10003, 0, NONE.0, 2, AI[2,0], 5V, 0V/0mA, 5V, 0.0, 5.0
...
REM Převod načtených hodnot na teploty ve °C
Ni1000U2T AI[0,0], Teploty[0,0], 6180ppm, 15.0, 3920.0
Ni1000U2T AI[1,0], Teploty[1,0], 6180ppm, 15.0, 3920.0
Ni1000U2T AI[2,0], Teploty[2,0], 6180ppm, 15.0, 3920.0
...
```

9.4. Nástěnné ovladače NOAxx

Tyto ovladače slouží např. k měření teploty prostoru, nastavení požadavku korekce teploty, nastavení režimů KOMFORT/ÚTLUM, případně dalších hodnot, dle typu ovladače. Korekce u ovladače **NOA21** a **NOA25** se nastavuje otočným kolečkem, u ovladače **NOA23** pomocí tlačítka a sloupce LED.

Rozdíl v nastavování parametrů komunikace (adresa, rychlost, zakončení RS485), oproti rozšiřujícím modulům, může být pouze v použití propojek namísto přepínačů. Význam propojek, včetně dalších informací o ovladačích naleznete v příslušném katalogovém listu.

9.4.1 Popis I/O struktury ovladačů NOA2x

Ovladače **NOA2x** se definují jako 4 uzly (AI, AO, DI a DO), se shodně nastavenou adresou. Pouze jeden z těchto uzlů může mít nastavenou hodnotu parametru Kontrola, jinak by mohlo docházet k problémům s navazováním komunikace.

Uzel AI:
Kanál 0: Měřená teplota -55 až 125 °C
Kanál 1: Korekce -100 až 100 %, u **NOA23** se nastavuje pouze v krocích -100, -50, 0, 50, 100

Uzel DI a DO:
Bit 0 – Attention
Bit 1 – KOMFORT
Bit 2 – ÚTLUM
Bit 3 – AUTO
Bit 4 – I
Bit 5 – II
Bit 6 – III

Ovladač nastavuje vždy pouze 1 bit z bitů 1 .. 6. Kromě stavu po resetu (připojení napájecího napětí), kdy jsou všechny bity rovny 1 – indikace stavu po resetu pro MASTERa. Zapíše-li MASTER do DO nesmyslnou kombinaci bitů (KOMFORT+ÚTLUM, ÚTLUM +II, apod.), ovladač zapsanou hodnotu ignoruje a v DI vrací původní hodnotu.

Bit Attention ovladač nastavuje:
- po resetu
- po ukončení editace (3 sekundy po posledním stisku tlačítka, viz níže)

MASTER nastavuje bit Attention:
- při inicializaci ovladače po resetu
- jako oznámení ovladači, že vzal v úvahu výsledek editace

Nastavení bitu Attention v DO způsobí vynulování téhož bitu v DI.

Uzel AO:
Kanál 0: Korekce (jen **NOA23**)
Kanál 1: nevyužito
Kanál 2: Jas LED

Ovladač **NOA23** zaokrouhlí přijatou hodnotu korekce na celý nejbližší násobek velikosti kroku (-100, -50, 0, 50, 100) a v AI vrací takto zaokrouhlenou hodnotu.
Ovladače bez možnosti nastavení korekce tuto hodnotu ignorují.

V režimu editace (vstup stiskem tlačítka) ovladač **NOA2x** ignoruje zaslané DO i korekci. Režim editace skončí a k návratu do normálního režimu (s možností vnucení DO+korekce MASTERem) dojde ve třech krocích:
- prodleva tři sekundy od posledního stisku tlačítka
- nastavení bitu Attention ze strany **NOA2x** (v DI)
- vynulování bitu Attention MASTERem (nastavením Attention v DO do 1)

Teprve v okamžiku provedení posledního kroku je možno znovu vnucovat DO+korekci.
Hodnota jasu se zpracovává nezměněně i v režimu editace.

Hodnota jasu LED je rozdělena na 5 kroků. V uvedeném příkladě pro **NOA2x** ji lze nastavovat v rozsahu 0 až 4, přičemž při hodnotě 0 je maximální jas LED a při hodnotě 4 jsou LED zhasnuté (svítí jen v režimu editace).

10. Dodatek B - Produkty jiných výrobců

Dostupné rozšiřující moduly jiných výrobců

Výrobce	Typ	Popis
COMET SYSTEM, s.r.o.	T3411 T3418 atd.	Průmyslové a interiérové snímače teploty, vlhkosti, tlaku. Úplný přehled produktů je na stránkách výrobce. Komunikace jako s klasickým AI modulem. www.cometsystem.cz
JEVEL Elektro	Z0J2	Univerzální 2 × 5 znakový sedmisegmentový displej. Komunikace jako s klasickým DO modulem.
REGMET	P1xA	Snímače teploty. Měření se provádí vestavěným čidlem Ni1000. Měřenou hodnotu vyhodnocuje elektronika umístěná v hlavici snímače. Snímače jsou kalibrovány v rozsahu teplot -30 .. 150 °C. Komunikace jako s klasickým AI modulem. www.regmet.cz
	ARDx	Teploměry a ovladače tohoto typu jsou určeny ke snímání a případně ke korekci prostorové teploty interiéru. Měření se provádí vestavěným čidlem Ni1000 zabudovaným do předního panelu. Snímač je kalibrován pro rozsah teplot -30 .. 60 °C. Komunikace jako s klasickým AI modulem. www.regmet.cz
SENSIT s.r.o.	SD11x	Snímače řady SD 11x – inteligentní snímače teploty pro venkovní prostředí. Komunikace jako s klasickým AI modulem. www.sensit.cz
SMARIS s.r.o.	6×AR	Univerzální vícevstupý programovatelný převodník. Každému ze vstupů lze programově nastavit měřící rozsah, požadavek na linearizaci vstupního signálu a konstantu tlumení. Komunikace jako s klasickým AI modulem. www.smaris.cz

Poznámka

Tento seznam modulů je aktuální k datu vytvoření této aplikační poznámky. Aktuální přehled všech dostupných modulů naleznete na výše uvedených webových stránkách.

11. Technická podpora

Veškeré informace ohledně komunikace v síti ARION Vám poskytne oddělení technické podpory firmy AMiT. Technickou podporu můžete kontaktovat nejlépe prostřednictvím emailu na adrese **support@amit.cz**.

12. Upozornění

AMiT spol. s r.o. poskytuje informace v tomto dokumentu, tak jak jsou, nepřejímá žádné záruky, pokud se týče obsahu tohoto dokumentu a vyhrazuje si právo měnit obsah dokumentu bez závazku tyto změny oznámit jakékoli osobě či organizaci.

Tento dokument může být kopírován a rozšiřován za následujících podmínek:

1. Celý text musí být kopírován bez úprav a se zahrnutím všech stránek.

2. Všechny kopie musí obsahovat označení autorského práva společnosti AMiT spol. s r. o. a veškerá další upozornění v dokumentu uvedená.

3. Tento dokument nesmí být distribuován za účelem dosažení zisku.

V publikaci použité názvy produktů, firem apod. mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.