

AIBUS 2

specifikace
komunikačního
protokolu

Obsah

1.	Úvodní popis	
1.1.	Charakteristika	I - 1
2.	Základní popis struktury protokolu	
2.1.	Úvod	I - 2
2.2.	Fyzická vrstva protokolu	I - 2
2.3.	Linková vrstva protokolu	I - 3
2.4.	Terminologie	I - 3
3.	Programátorský model jednotky	
3.1.	Úvod	I - 5
3.2.	Externí periferie s přímým přístupem	I - 5
3.3.	Externí adresovatelné periferie	I - 6
3.4.	Interní periferie	I - 6
4.	Popis protokolu	
4.1.	Úvod	I - 7
4.2.	Popis zprávy "master - slave"	I - 7
4.3.	Popis zprávy "slave - master"	I - 8
4.4.	Přímý příkaz	I - 8
4.5.	Status registr	I - 9
4.6.	Časování sběrnice	I - 9
4.7.	Periferie	I - 10
5.	Číselné formáty	
5.1.	Úvod	I - 11
5.2.	Celočíselný formát	I - 11
5.3.	Formát s plovoucí desetinnou čárkou	I - 11
6.	Popis funkcí	
6.1.	Úvod, seznam funkcí	I - 12
6.2.	FN 0 - čtení externí periferie s přímým přístupem	I - 12
6.3.	FN 1 - zápis do externí periferie s přímým přístupem	I - 12
6.4.	FN 2 - čtení externí adresovatelné periferie	I - 13
6.5.	FN 3 - zápis do externí adresovatelné periferie	I - 13
6.6.	Funkce pro obsluhu interních periferií (FN 100-103)	I - 13
6.7.	FN 200 - čtení speciálních registrů	I - 14

Přílohy:

Příloha II - CRC 16

Příloha III - Aktuální seznam periferií jednotek Tedia MICROUNIT SERIE

1. Úvod

1.1. Charakteristika

Komunikační protokol AIBus-2 je určen pro vytváření sítí typu "Master - Slave", umožňuje však i základní přenos dat mezi dvěma zařízeními. Vyznačuje se jednoduchou, avšak spolehlivou a úspornou strukturou, pro kterou je charakteristické:

- "master - slave" topologie s jedním "masterem" na lince ("multimaster" varianta není protokolem specifikována)
- pevná délka přenášených povelů
- zpětné potvrzování každého povelu
- rozlišení směru přenosu proměnným paritním bitem (není nutné u duplexního vedení)
- identifikace koncového zařízení 8 bitovou adresou
- předávání diagnostických informací prostřednictvím chybové proměnné
- kontrola bezchybného přenosu zprávy pomocí 16-bitového CRC

2. Základní popis struktury protokolu

2.1. Úvod

Ve světě je řada výrobců komponent pro počítačové a průmyslové systémy, základní stanovení zásad jejich vzájemné slčitelnosti je pak shrnuto v normě IEEE 802. Tato norma v obecné podobě určuje způsob komunikace mezi stanicemi od úrovně fyzického připojení až po přenos programů či dat. Normu pro vzájemnou komunikaci vydala ISO pod označením "Open System Interconnection" (OSI).

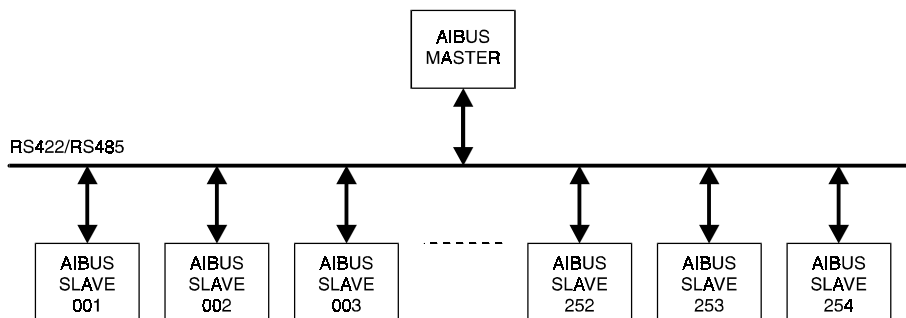
Protokol ve zmíněné normě je rozdělen na následujících sedm vrstev:

fyzická vrstva	definuje přenosové médium, konektory, kódování, topologii, ...
linková vrstva	definuje způsob přenášení zpráv v síti a jejich zajištění proti chybám a ztrátě informace
síťová vrstva	definuje způsob pohybu datových paketů v síti z hlediska přepojovacích uzlů
transportní vrstva	definuje adresaci zařízení a aplikačních programů v síti
relační vrstva	vytváří logické rozhraní pro aplikační programy využívající služeb sítě
presentační vrstva	transformuje data do přenosové formy, provádí konverzi kódů, kompresi či šifrování dat, ...
aplikační vrstva	charakterizuje aplikační programy využitě k podpoře sítě

Protokol **AIBUS-2**, právě jako řada dalších jednoduchých protokolů, má pevně definovanou linkovou vrstvu a částečně pak i vrstvu fyzickou. Zbývající vrstvy nejsou specifikovány a jsou zcela závislé na konkrétním technickém zařízení.

2.2. Fyzická vrstva protokolu

Přestože komunikační protokol **AIBUS-2** není zcela pevně vázán na přenosové médium, za nejběžnější lze označit dvou vodičové poloduplexní vedení standardu RS485. Z hlediska topologie sítě se pak jedná o typickou sběrníkovou architekturu "master-slave" (viz. Obr.1.)




Obr.1. Topologie sítě.

V praxi lze protokol implementovat prakticky na jakémkoliv rozhraní, které splňuje dále uvedené vlastnosti:

- topologie sítě "master - slave"
- asynchronní sériový duplexní nebo poloduplexní přenos
- přenos 8-bitových binárních dat, 1 stopbit, sudá i lichá parita (viz pozn.)

Kromě zmíněného rozhraní RS485 lze jako přenosového média využít rozhraní RS422, RS232, proudovou smyčku 20mA (v duplexním i poloduplexním zapojení), optická vlákna apod.

 *Paritní bit je využit pro signalizaci směru přenosu dat, v případě plně duplexního vedení postačuje podpora sudé parity nebo podpora parity zcela vypustit.*

2.3. Linková vrstva protokolu

Komunikační protokol AIBUS-2 pro svoji topologii "master - slave" využívá adresace koncových zařízení (tzn. "slave" jednotek) prostřednictvím 8-bitové adresy; na síti tedy může kromě řídicího systému ("masteru") současně spolupracovat maximálně 256 jednotek. Praktická realizace však tento počet omezuje o dvě vyhrazené (0 a 255) na konečných 254.

Narozdíl k počtu "slave" jednotek protokol AIBUS-2 předpokládá řízení jediným nadřízeným systémem "master"; žádná pravidla pro sdílení sítě více mastery nejsou specifikována.

Přenos dat je organizován ve zprávách konstantní délky se základní ochranou každého znaku paritním bitem. Vzhledem k binárnímu charakteru dat je každá zpráva zajištěna 16-bitových CRC kódem.

Základní struktura přenosové zprávy je znázorněna na obrázku Obr.2.

<i>adresa</i>	<i>funkce</i>	<i>periferie</i>	<i>příkaz / status</i>	<i>datové pole</i>	<i>CRC16</i>
<i>1 byte</i>	<i>1 byte</i>	<i>1 byte</i>	<i>1 byte</i>	<i>4 byte</i>	<i>2 byte</i>

Obr.2. *Struktura zprávy.*

K rozlišení směru přenosu dat u poloduplexních vedení je využito proměnného paritního bitu; výzva (tzn. zpráva ve směru "master - slave") je přenášena s paritou sudou, odpověď (tzn. zpráva ve směru "slave - master") s paritou lichou. V případě duplexních vedení lze alternativně využít sudé parity v obou směrech přenosu, popř. ochranu paritním bitem vypustit.

2.4. Terminologie

V dalším popisu budou využívány tyto pojmy:

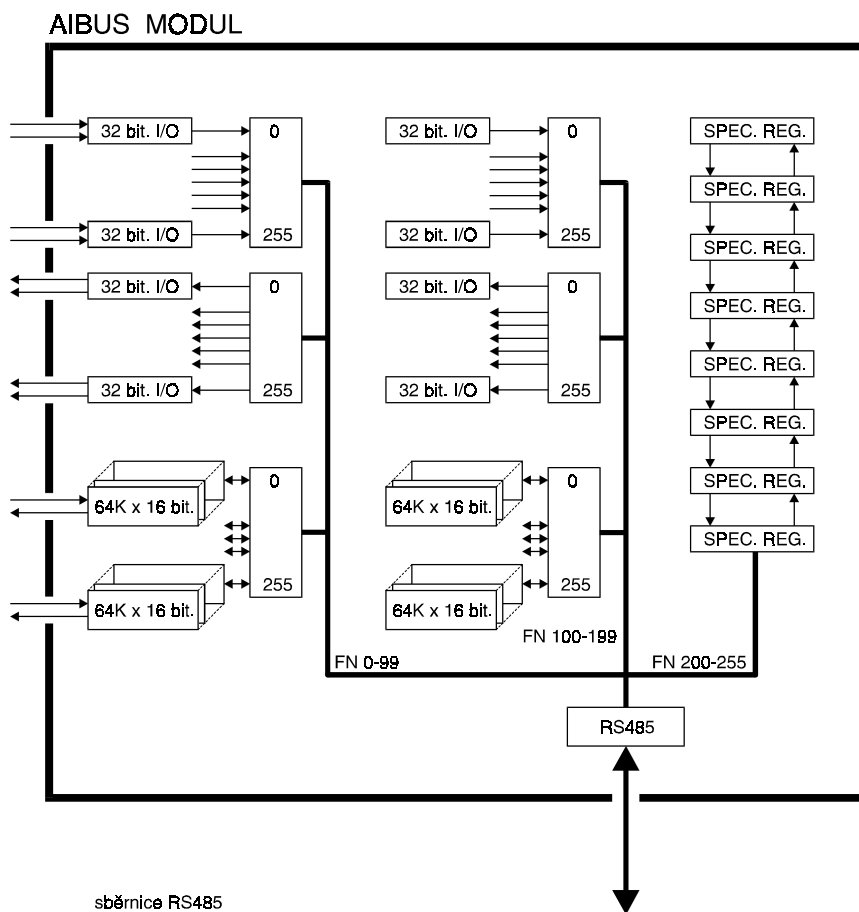
master	nadřízený počítač (např. PC), je iniciátorem přenosu
slave	podřízená jednotka, pouze odpovídá na vyslanou zprávu
jednotka	zařízení připojené k síti charakterizované jednou adresou

výzva	zpráva ve směru "master - slave"
odpověď	zpráva ve směru "slave - master"
adresa	jedinečné identifikační číslo jednotky na síti
funkce	operace prováděná s periferií jednotky (čtení dat, zápis dat, ...)
periferie	funkční blok jednotky (A/D převodník, D/A převodník, čítač, řadič DIO portů, ...)
příkaz/status	informace přenášená v obou směrech, význam nezávislý na ostatních parametrech zprávy
datové pole	obsahuje datovou informaci pro specifikovanou funkci
CRC16	slouží k zabezpečení všech předešlých znaků zprávy

3. Programátorský model jednotky

3.1. Úvod

Pro vytvoření ucelené představy o vlastnostech protokolu a významu jednotlivých funkcí je v této kapitole uvedena obecná interní struktura jednotky.



Obr.3. Vnitřní struktura jednotky.

Jak vyplývá z uvedeného obrázku, programátorský model jednotky předpokládá dva oddělené prostory periférií - externí a interní. Oba prostory pak mají deklarované periférie s přímým přístupem a periférie adresovatelné. Externí a interní prostory jsou z hlediska programové podpory zcela analogické; odlišné jsou pouze funkce, sloužící k jejich obsluze. Model dále předpokládá existenci speciálních registrů pro konfigurační a identifikační data.

3.2. Externí periférie s přímým přístupem

Typickým zástupcem periférie s přímým přístupem jsou digitální porty; datové pole přenáší informaci ve 32-bitovém formátu; každý bit pak představuje jeden z 32

digitálních vstupů nebo výstupů. Obdobným způsobem je však řešena i obsluha analogových vstupů nebo čítačů (1 periférie = jeden AIN nebo 1 čítač).

Protokol tedy umožňuje podporu 256 vstupních periférií a 256 periférií výstupních, což v praxi odpovídá např. 8192 digitálním vstupům a 8192 výstupům.

3.3. Externí adresovatelné periférie

Typickým zástupcem adresovatelné periférie je paměť RAM, EEPROM a další; datové pole pak současně přenáší adresu paměťového prostoru (2 byte) a datovou informaci (2 byte). Kromě periférií paměťového charakteru je obdobným způsobem řešena obsluha složitějších obvodů, např. modulu PID regulátoru nebo sběrnice.

Protokol tedy umožňuje podporu 256 adresovatelných periférií, což v praxi odpovídá celkovému adresovému prostoru 32MB.

3.4. Interní periférie

Obdobou externích periférií jsou periférie interní, narozdíl však od předešlých neslouží k ovládání provozních funkcí, ale zpravidla ke konfiguraci jednotky, přenosu stavových informací nebo jejich nastavení, řízení modulu reálného času (RTC) apod. Jejich struktura i rozsah adresového prostoru je totožný s perifériemi externími.

4. Popis protokolu

4.1. Úvod

Jak vyplynulo z druhé kapitoly, definice protokolu je rozdělena do jednotlivých vrstev; z nichž AIBUS-2 má pevně definovanou linkovou vrstvu a částečně pak i vrstvu fyzickou.

Další text se již bude týkat výhradně popisu linkové vrstvy protokolu, tzn. popisu struktury zprávy a v páté kapitole pak implementovaných funkcí zajišťující obsluhu periferních obvodů jednotky.

4.2. Popis zprávy "master - slave"

Zpráva "master - slave", jinak nazývaná také výzvou, inicializuje přenos dat na lince a je vysílána se sudou paritou.

Struktura je zakreslena na obrázku Obr.4.

adresa	funkce	periferie	přímý příkaz	datové pole	CRC16
1 byte	1 byte	1 byte	1 byte	4 byte	2 byte

Obr.4. Struktura vysílané zprávy.

Zpráva je rozdělena na tyto části:

Adresa	identifikuje koncové zařízení, tzn. "slave" jednotku. Adresový prostor jednotek je oproti plnému počtu 256 omezen o dvě vyhrazené adresy. Adresa "0" je určena pro úvodní inicializaci jednotky, adresa "255" pak umožňuje předávání společných příkazů současně všem jednotkám bez ohledu na jejich adresu (např. příkaz strobování analogových vstupů, stavu čítačů, DIO portů).
Funkce	představuje operaci, která bude provedena s daty na příslušném periferním obvodu (např. zápis dat).
Periferie	je parametr specifikující funkční obvod "slave" jednotky (např. řadič DIO), kterému je přiřazena následující funkce a data.
Přímý příkaz	slouží k ovládní některých vyhrazených funkcí bez ohledu na právě prováděnou funkci.
Datové pole	přenáší vlastní informaci pro funkci a periferii. V případě přenosu informace ve směru "master - slave" (např. zápis dat na DIO port) obsahuje datové pole příslušná data. Jedná-li se naopak o přenos informace ve směru opačném (např. čtení dat z DIO portu), je jeho obsah nevýznamný.
CRC16	je ochranný kód, sloužící k zabezpečení všech předešlých znaků zprávy. Popis použitého cyklického kódu s generačním polynomem délky 16 je uveden v příloze příručky.

4.3. Popis zprávy "slave - master"

Zpráva "slave - master", jinak nazývaná také odpovědí, je reakcí na inicializační výzvu a je vysílána s lichou paritou.

Struktura je zakreslena na obrázku Obr.5.

adresa	funkce	periferie	status	datové pole	CRC16
1 byte	1 byte	1 byte	1 byte	4 byte	2 byte

Obr.5. Struktura přijímané zprávy.

Zpráva je rozdělena na tyto části:

Adresa	má totožný význam jako u předešlé výzvy, tzn. informuje "master" o odpovídající jednotce.
Funkce	má totožný význam jako u předešlé výzvy, tzn. informuje "master" o provedené funkci odpovídající jednotky.
Periferie	má totožný význam jako u předešlé výzvy, tzn. informuje "master" o periférii odpovídající jednotky.
Status	slouží k přenosu významných informací (např. informace o chybových stavech jednotky) bez ohledu na prováděnou funkci nebo zvolenou periférii.
Datové pole	přenáší vlastní informaci pro funkci a periférii. V případě přenosu informace ve směru "slave - master" (např. čtení dat z DIO portu) obsahuje datové pole příslušná data. Jedná-li se naopak o přenos informace ve směru opačném (např. zápis dat na DIO port), je jeho obsah nevýznamný.
CRC16	má totožný význam jako u předešlé výzvy.

4.4. Přímý příkaz

Přímý příkaz slouží k ovládní prioritních funkcí jednotky a je obsažen v každé přenášené výzvě. Struktura tohoto registru je zakreslena na obrázku Obr.6.

7.	6.	5.	4.	3.	2.	1.	0.
rezerva	rezerva	rezerva	rezerva	rezerva	rezerva	rezerva	SMPL

Obr.6. Struktura registru přímého příkazu.

SMPL	příkaz k zavzorkování vstupních dat jednotky (určeno pro synchronní záznam více vstupních hodnot, podrobný popis uveden v příručce příslušné jednotky)
rezerva	dosud neobsazený bit

4.5. Status registr

Jednotky poskytují základní chybové a provozní informace ve zvláštní proměnné označované jako stavový registr. Tato 32-bitová proměnná je přístupná (pro čtení i zápis) jako interní přímo adresovaná periferie prostřednictvím funkcí 100/101.

Nejvýznamnějších 8 bitů této proměnné (význam společný pro všechny jednotky) je přenášeno v každé odpovědi pod označení Status Registr. Struktura stavového registru je znázorněna na obrázku Obr.7.

7.	6.	5.	4.	3.	2.	1.	0.
rezerva	ERR24	RST	PWR	rezerva	PRF	FN	SMPL

Obr.7. Struktura stavového registru.

Význam jednotlivých bitů (všechny aktivní v logické úrovni H):

ERR24	24-bitová stavová proměnná obsahuje alespoň jeden aktivní příznak (význam těchto příznaků není standardizován)
RST	příznak indikující stav po signálu reset
PWR	příznak indikující stav po výpadku napájecího napětí
PRF	neznámá periferie; příznak signalizující, že funkce požadovala obsluhu periferie, kterou jednotka neobsahuje
FN	příznak signalizující, že požadovaná funkce nebyla provedena (např. zápis do nepovolené oblasti EEPROM)
SMPL	signalizace zavzorkovaných dat (viz 4.4. Příímý příkaz)
rezerva	dosud neobsazený příznak

4.6. Časování sběrnice

Pro definici časových poměrů lze uvažovat tyto parametry:

- prodleva mezi jednotlivými znaky výzvy
- doba provedení funkce (interval od konce výzvy do prvního znaku odpovědi)
- prodleva mezi jednotlivými znaky odpovědi
- doba na zotavení jednotky (schopnost jednotky přijmout novou výzvu)

Tolerance prodlevy mezi jednotlivými znaky patří mezi důležité parametry ovlivňující spolehlivost přenosu. Z tohoto důvodu jednotky trvale vyhodnocují kontinuitu přijímaných dat a přerušení výzvy o délce větší než 2 znaky považuje za nekorektní ukončení. Nadřazený systém tedy musí výzvu vyslat bez přerušení.

Druhý uvedený interval, tedy doba k provedení funkce, je závislý na charakteru funkce, typu a konfiguraci jednotky. Běžné provozní funkce jednotek DIO portů jsou prováděny bez prodlení, odpověď je odesílána se zpožděním pod 1ms. Naopak jednotky či funkce, které provádějí obsluhu pomalých periferních obvodů, mohou tento časový interval prodloužit až k hranici 20 ms. Přesná specifikace tohoto parametru je tedy uváděna u popisu jednotlivých jednotek.

Pro specifikaci třetího uvedeného parametru, tedy prodlevu mezi jednotlivými znaky odpovědi, platí obdobné limity jako pro parametry výzvy.

Doba potřebná k zotavení jednotky z prováděné funkce je závislá na charakteru funkce, resp. typu jednotky, proto je uváděna při popisu jednotlivých jednotek.

U naprosté většiny jednotek a jejich provozních funkcí je tato doba nulová; jednotka je tedy připravena k přijímání nové výzvy ihned po dokončení odpovědi.

4.7. Periferie

Přestože shodné řazení periférií u různých typů jednotek není specifikací prokolu pevně definováno, je unifikace výhodná pro přenositelnost software mezi moduly.

Z tohoto důvodu jsou jednotlivé periferie jednotek Tedia zařazovány tak, aby stejná funkce (např. obsluha DIO portů) byla řízena prostřednictvím stejné periferie.

Aktuální seznam periferních obvodů je uveden v příloze příručky.

5. Číselné formáty

5.1. Úvod

Protokol má definovány dva základní číselné formáty; celočíselný a s plovoucí desetinnou čárkou. V případě potřeby však konkrétní jednotka může mít definován vlastní datový formát.

5.2. Celočíselný formát

Tento typ číselného formátu využívají periferie typu čítač, přímý výstup A/D převodníku apod.; je však prakticky využit i pro přenos dat digitálních portů.

Struktura celočíselného formátu je znázorněna na Obr.8.

<i>celočíselný údaj v rozsahu 0 ... 2³² -1</i>			
<i>D31...D24</i>	<i>D23...D16</i>	<i>D15...D8</i>	<i>D7...D0</i>

Obr.8. *Struktura celočíselného formátu.*

5.2. Formát s plovoucí desetinnou čárkou

Tento číselný formát je využíván např. pro přenos kalibrovaných dat z analogových vstupů, popř. jiných periférií analogového charakteru s přepočtem na skutečnou fyzikální veličinu.

Struktura tohoto formátu je znázorněna na Obr.9.

<i>PE</i>	<i>exponent</i>	<i>PM</i>	<i>mantisa v rozsahu 0 ... 2²³ -1</i>		
<i>D31</i>	<i>D30...D24</i>	<i>D23</i>	<i>D22...D16</i>	<i>D15...D8</i>	<i>D7...D0</i>

Obr.9. *Struktura formátu s plovoucí desetinnou čárkou.*

Význam jednotlivých proměnných:

mantisa	23-bitové číslo (bez desetinné tečky)
PM	znaménkový bit mantisy ("1" představuje záporné číslo)
exponent	7-bitové číslo (bez desetinné tečky)
PE	znaménkový bit exponentu (význam jako PM)

příklad: (1 0000100 0 0000001 00000000 00000001) = 65537 * 10⁻⁴



Datové formáty AIBus-2 vycházejí z doporučení IEEE 488.2.

6. Popis funkcí

6.1. Úvod, seznam funkcí

Funkce jsou předdefinovány na:

FN0 ... FN99	funkce pro obsluhu externích periférií jednotky
FN100 ... FN199	funkce pro obsluhu interních periférií
FN200 ... FN249	speciální funkce společné pro všechny jednotky
FN250 ... FN255	speciální servisní a diagnostické funkce (nejsou dostupné v běžném provozu)

6.2. FN 0 - čtení externí periferie s přímým přístupem

Tato funkce je určena pro čtení dat z analogových vstupů, čítačů, logických portů a dalších periférií, které pracují s datovou strukturou s rozsahem do 32 bitů. Struktura funkce je zakreslena na obrázku Obr.10.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
ADR	0	PRF	DIR	??	??	??	??	CRC16	
ADR	0	PRF	STAT	D0	D1	D2	D3	CRC16	

Obr.10. Struktura funkce 0.

Protože datové pole výzvy nepřenáší žádnou informaci, na jeho obsahu nezáleží a je jednotkou ignorován. Datové pole odpovědi obsahuje 32-bitová data přenášená v pořadí od nejnižšího byte po nejvyšší. Význam dat je závislý na typu periferie.

6.3. FN 1 - zápis do externí periferie s přímým přístupem

Tato funkce je určena k přenosu dat pro analogové výstupy, logické porty, programování čítačů a dalších periférií, které pracují s datovou strukturou s rozsahem do 32 bitů. Struktura funkce je zakreslena na obrázku Obr.11.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
ADR	1	PRF	DIR	D0	D1	D2	D3	CRC16	
ADR	1	PRF	STAT	??	??	??	??	CRC16	

Obr.11. Struktura funkce 1.

Datové pole výzvy obsahuje 32-bitová data přenášená v pořadí od nejnižšího byte po nejvyšší. Význam dat je závislý na typu periferie. Odpověď jednotky žádná platná data nepřenáší, všechny čtyři byte jsou nulové.

6.4. FN 2 - čtení externí adresovatelné periferie

Tato funkce je určena k přenosu dat pro periferie adresovatelného charakteru; typickým představitelem jsou paměťové bloky RAM nebo EEPROM. Struktura funkce je zakreslena na obrázku Obr.12.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
ADR	2	PRF	DIR	A0	A1	??	??	CRC16	
ADR	2	PRF	STAT	A0	A1	D0	D1	CRC16	

Obr.12. Struktura funkce 2.

V datovém poli výzvy je přenášena 16-bitová adresa pro periferii, ze které má být provedena operace čtení dat. Dva volné byte neobsahují žádná data a jejich obsah je jednotkou ignorován.

Datové pole odpovědi obsahuje 16-bitovou adresu (je totožná s adresou ve výzvě) a dále 16-bitová data načtená z periferie. Význam dat je závislý na typu periferie.

V obou případech je adresa i data přenášena v pořadí od nejnižšího byte po nejvyšší.

6.5. FN 3 - zápis do externí adresovatelné periferie

Tato funkce je určena k přenosu dat pro periferie adresovatelného charakteru; typickým představitelem jsou paměťové bloky RAM nebo EEPROM. Struktura funkce je zakreslena na obrázku Obr.13.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
ADR	3	PRF	DIR	A0	A1	D0	D1	CRC16	
ADR	3	PRF	STAT	??	??	??	??	CRC16	

Obr.13. Struktura funkce 3.

V datovém poli výzvy jsou přenášena 16-bitová adresa a 16-bitová data pro určenou periferii. Význam dat je závislý na typu periferie.

Datové pole odpovědi žádná platná data nepřenáší, všechny čtyři byte jsou nulové.

V obou případech je adresa i data přenášena v pořadí od nejnižšího byte po nejvyšší.

6.6. Funkce pro obsluhu interních periférií (FN 100-103)

Funkce 100 - 103 jsou analogické funkcím 0 - 3 včetně zcela shodné struktury.

Odlišností je pouze charakter periferie, k jejíž obsluze jsou určeny. Podrobně viz třetí kapitola - Programátorský model jednotky.

6.7. FN 200 - čtení speciálních registrů

Funkce 200 je určena pro čtení identifikačních dat jednotky, ke kterým patří zejména název jednotky, verze firmware a další. Struktura funkce je zakreslena na obrázku Obr.14.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
ADR	200	PRF	DIR	??	??	??	??	CRC16	
ADR	200	PRF	STAT	D0	D1	D2	D3	CRC16	

Obr.14. Struktura funkce 200.

Protože datové pole výzvy nepřenáší žádnou informaci, na jeho obsahu nezáleží a je jednotkou ignorován. Datové pole odpovědi obsahuje 32-bitová data přenášená v pořadí od nejnižšího byte po nejvyšší. Význam dat je závislý na typu periferie

1. Úvod

1.1. Charakteristika

Zabezpečení vyslaného paketu včetně řídicího pole a adresy je provedeno cyklickým kódem s generačním polynomem délky 16; proto také délka kontrolního slova CRC je 16 bitů. Způsob vyhodnocení chyby je pouze detekcí, pro korekční (tzn. samoopravné kódy) by byla nutná vysoká redundance, která z hlediska rychlosti přenosu a využití kanálu není opodstatněná. Proto se při chybě přijatého rámce přenos opakuje.

Pro zabezpečení byl využit generační polynom typu $x^{16} + x^{15} + x^2 + 1$, což odpovídá binárnímu vyjádření 1 1000 0000 0000 0101.


1.2. Algoritmus výpočtu

Algoritmus výpočtu bude demonstrován na příkladu kódování zprávy:

D1	D2	CRC16	
02 _H	07 _H	??	??

Postup výpočtu CRC16 (viz tabulka Tab.1.):

- 1) Do 16-bitového registru je vložena konstanta FFFF_H.
- 2) Provedena operace XOR s D1 a obsahem 16-bitového registru (3. řádek)
- 3) provedena rotace vpravo (zleva doplňována "0", vpravo přesun do příznaku)
- 4) Je-li příznak 1, pak provedena operace XOR s polynomem (5. řádek).
- 5) Kroky 3) a 4) jsou vykonávány, dokud není provedena rotace 8x.
- 6) Provedena operace XOR s D2 a obsahem 16-bitového registru (22. řádek)
- 7) Zopakován postup podle kroků 3) až 5).
- 8) Obsah 16-bitového registru je výsledným kódem CRC16.

 Při výpočtu je polynom využit v 16-bit. tvaru s opačným pořadím bitů (1010000000000001).

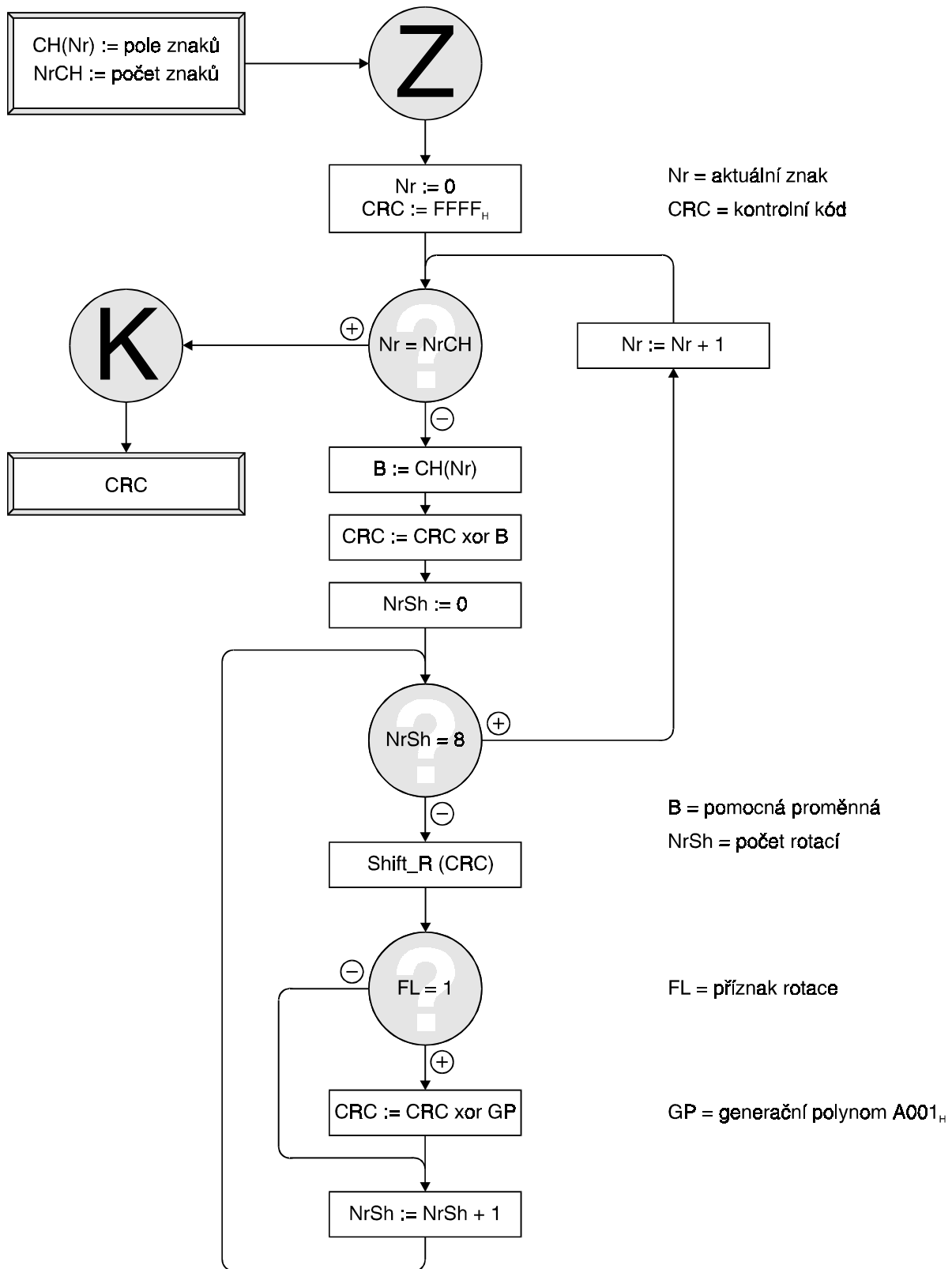
Výsledná zpráva bude odeslána ve tvaru:

D1	D2	CRC16	
02 _H	07 _H	41 _H	12 _H

 Algoritmus výpočtu CRC16 je dále demonstrován na vývojovém diagramu (viz Obr.1.).

ŘDK	operace	16 - bit. registr				příznak
1.	FFFF _H	1111	1111	1111	1111	
2.	D1 (02 _H)			0000	0010	
3.	(1.) xor (2.)	1111	1111	1111	1101	
4.	rotace vpravo	0111	1111	1111	1110	1
5.	polynom	1010	0000	0000	0001	
6.	(4.) xor (5.)	1101	1111	1111	1111	
7.	rotace vpravo	0110	1111	1111	1111	1
8.	polynom	1010	0000	0000	0001	
9.	(7.) xor (8.)	1100	1111	1111	1110	
10.	rotace vpravo	0110	0111	1111	1111	0
11.	rotace vpravo	0011	0011	1111	1111	1
12.	polynom	1010	0000	0000	0001	
13.	(11.) xor (12.)	1001	0011	1111	1110	
14.	rotace vpravo	0100	1001	1111	1111	0
15.	rotace vpravo	0010	0100	1111	1111	1
16.	polynom	1010	0000	0000	0001	
17.	(15.) xor (16.)	1000	0100	1111	1110	
18.	rotace vpravo	0100	0010	0111	1111	0
19.	rotace vpravo	0010	0001	0011	1111	1
20.	polynom	1010	0000	0000	0001	
21.	(19.) xor (20.)	1000	0001	0011	1110	
22.	D2 (07 _H)			0000	0111	
23.	(21.) xor (22.)	1000	0001	0011	1001	
24.	rotace vpravo	0100	0000	1001	1100	1
25.	polynom	1010	0000	0000	0001	
26.	(24.) xor (25.)	1110	0000	1001	1101	
27.	rotace vpravo	0111	0000	0100	1110	1
28.	polynom	1010	0000	0000	0001	
29.	(27.) xor (28.)	1101	0000	0010	1111	
30.	rotace vpravo	0110	1000	0001	0111	1
31.	polynom	1010	0000	0000	0001	
32.	(30.) xor (31.)	1100	1000	0010	0110	
33.	rotace vpravo	0110	0100	0001	0011	0
34.	rotace vpravo	0011	0010	0000	1001	1
35.	polynom	1010	0000	0000	0001	
36.	(34.) xor (35.)	1001	0010	0000	1000	
37.	rotace vpravo	0100	1001	0000	0100	0
38.	rotace vpravo	0010	0100	1000	0010	0
39.	rotace vpravo	0001	0010	0100	0001	0
40.	CRC16	12 _H		41 _H		

Tab.1. Praktický příklad výpočtu CRC16.



Obr.1. Algoritmus výpočtu CRC16.

1. Periferie modulů Tedia

1.1. Úvod

Příloha obsahuje přehled použitých periférií určených pro uživatelskou obsluhu. Moduly mají zpravidla implementovány servisní funkce a periferie určené výhradně pro diagnostické účely a jejich seznam není proto zveřejňován.

Seznam periférií může být volně rozšiřován v souladu se specifikací protokolu a ohledem na předdefinované standardní funkce. Níže uvedený seznam proto odráží aktuální stav.

1.2. Seznam periférií

Moduly Tedia používají (stav srpen 1997) externí periferie s přímým přístupem a interní periferie adresovatelné i s přímým přístupem. Speciální a diagnostické funkce pak podporují zbývající obvody.

funkce	periferie	operace	význam
0 / 1	0	RD / WR	analogový kanál 0 (vstupní / výstupní port)
0 / 1	1 ÷ 14	...	analogový kanál 1÷14 (vstupní / výstupní port)
0 / 1	15	RD / WR	analogový kanál 15 (vstupní / výstupní port)
0	16	RD	čidlo studeného konce TC
0 / 1	64	RD / WR	32-bitový digitální port 0 (vstupní / výstupní port)
0 / 1	65	RD / WR	32-bitový digitální port 1 (vstupní / výstupní port)
0 / 1	80	RD / WR	32-bitový čítač 0 (čtení / nastavení čítače)
0 / 1	81 ÷ 95	...	32-bitový čítač 1÷15 (čtení / nastavení čítače)
100 / 101	0	RD / WR	STATUS registr
100 / 101	1	RD / WR	RTC - čas
100 / 101	2	RD / WR	RTC - datum
100 / 101	3	RD	začátek výpadku napájecího napětí - čas
100 / 101	4	RD	začátek výpadku napájecího napětí - datum
100 / 101	5	RD	konec výpadku napájecího napětí - čas
100 / 101	6	RD	konec výpadku napájecího napětí - datum
102 / 103	0	RD / WR	EEPROM - konfigurační data
102 / 103	1	RD / WR	EEPROM - uživatelská data
200	0, 1, 2	RD / WR	identifikace modulu (typ, verze firmware)
250 - 255	-----	RD / WR	servisní a diagnostické funkce

Tab.1. Přehled periférií modulů Tedia.

