

System WACO

Petr Volný

Obsah

1	WACO protokol	1
1.1	Fyzická vrstva	1
1.1.1	RF parametry	1
1.1.1.1	Radiové parametry	1
1.1.1.2	Paketové parametry	2
1.2	Linková vrstva	2
1.2.1	Položka Dst	3
1.2.2	Položka Src	3
1.2.3	Položka Port	3
1.2.4	Položka Info	4
1.2.5	Položka Tid	4
1.3	Aplikační vrstva	4
1.3.1	Aplikace PING	5
1.3.2	Aplikace VIRTCHAN	5
1.3.3	Aplikace SISA	5
1.3.4	Aplikace NEP a SLIP	5
1.3.5	Aplikace WOR	6
1.3.6	Aplikace HELLO	6
1.4	SLIP protokol v RF paketech	6
1.5	Adresace systému WACO	6
1.6	Funkce opakovače	7
1.7	Použití bran	7
2	NEP protokol	9
2.1	Definice významu proměnných	10
2.2	NEP zpráva	10
2.2.1	Typy zpráv	10
2.2.1.1	Zpráva typu TRAP	11
2.2.1.2	Zprávy typu SET, GET, RESPONSE a DATA/INFO	11
2.2.1.3	Zpráva typu GETNEXT	12
2.2.2	Kódování dat aplikační vrstvy	12
2.2.2.1	O Objekt	13
2.2.2.2	T Typ a L Délka	13
2.2.2.3	V hodnota proměnné (value)	14
2.3	Kódy alarmů	15
2.3.1	Kód 0, RESET systému	15

2.3.2	Kód 1, Změna konfigurace	15
2.3.3	Kód 2, Příliš mnoho alarmů	16
2.3.4	Kód 10, Příliš nízká teplota	16
2.3.5	Kód 11, Příliš vysoká teplota	16
2.3.6	Kód 12, Teplota v normálu	16

Seznam tabulek

1.1	Základní parametry RF transceiveru	2
1.2	Paketové parametry RF transceiveru	2
2.1	Tabulka typů NEP proměnných	14

Seznam obrázků

1.1 Tvar radiového paketu	2
-------------------------------------	---

Kapitola 1

WACO protokol

WACO protokol je proprietární protokol firmy Softlink s.r.o. pro komunikaci radiových stanic.

Protokol umožňuje komunikovat stanicím prostřednictvím radiové sítě. Systém je koncipován tak, aby neměl žádný řídicí prvek, a tedy aby vyřazením libovolné stanice nebyla narušena schopnost komunikovat mezi ostatními stanicemi. Systém obsahuje prvek opakovače, který umožňuje komunikovat i zařízením, které nemají přímou radiovou viditelnost.

Pro větší hustotu radiových sítí používá systém opakovačů (repeater, range extender), které realizují směrovací algoritmus, konkrétně záplavové směrováním, chráněné položkou hop count, která limituje životnost paketu v rámci sítě.

Rovněž lze použít rozdělení překrývajících se sítí do kanálů podle šířky pásma, s jakou pracuje fyzická vrstva. Tato šířka pásma je přímo úměrná přenosové rychlosti, kterou fyzická vrstva využívá.

1.1 Fyzická vrstva

1.1.1 RF parametry

RF parametry se dají rozdělit do dvou skupin.

- radiové parametry
- paketové parametry

1.1.1.1 Radiové parametry

Radiové parametry transceiverů v systému WACO určují vlastně fyzickou vrstvu komunikačního protokolu WACO ve smyslu, jak je zakódována logická **1** a **0**. Původní systém používá transceiver řady **CC1101**, proto mnoho parametrů a módů provozu vycházejí z vlastností tohoto obvodu. Pro zachování kompatibility je tedy nutné dodržet tyto parametry i při použití jiného transceiveru. Seznam těchto parametrů je uveden v tabulce (1.1).

Realizace těchto parametrů má zásadní vliv na to, zda jsou RF transceivery schopny se navzájem

Tab. 1.1: Základní parametry RF transceiveru

Parametr	Hodnota
Frekvenční rozsah	868.0 ... 868.6 MHz
Šířka kanálu	200 kHz
Počet kanálů	3
Nosná frekvence kanálu 0	868.1 MHz
Šířka pásma přijímače	100 kHz
Modulace	GFSK (GFSK2)
Deviace	20 kHz
Bitová rychlost	38400

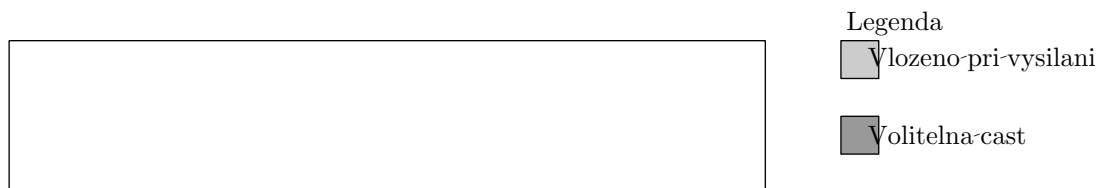
slyšet a rozkódovat příchozí bity paketu.

1.1.1.2 Paketové parametry

Paketové parametry určují tvar paketu, který transceiver vysílá a je schopen přijímat. Seznam parametrů udává tabulka (1.2).

Tab. 1.2: Paketové parametry RF transceiveru

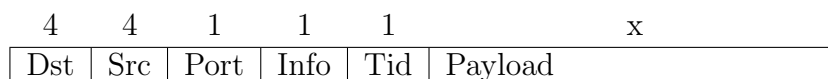
Parametr	Hodnota
Délka preamble	32 bitů
Počet synchronizačních byte	4
Synchronizační byty	0xD3 0x91 0xD3 0x91
Počet datových byte paketu	63 (+1 délka)
Data whitening	PN9 sequence
CRC zabezpečení	CRC-16 ($x^{16} + x^{15} + x^2 + 1$)
CRC inicializace	all 1's (0xFFFF)



Obr. 1.1: Tvar radiového paketu

1.2 Linková vrstva

Linková vrstva je tvořena paketem s následující strukturou:



Dst	4 byte adresy příjemce (unicast, multicast, broadcast)
Src	4 byte adresy odesílatele (unicast)
Port	identifikace aplikace
Info	maximální počet skoků, typ zařízení, šifrování
Tid	(transaction identifier) číslo transakce, pro potvrzování zpráv, potvrzovací bit (ACK)
Payload	data síťové vrstvy / aplikační data

Struktura v jazyce „C“ je následující:

```
typedef struct {
    uint32_t dstAddr;
    uint32_t srcAddr;
    uint8_t port;
    uint8_t info;
    uint8_t tid;
    uint8_t msg[MAX_DATA_LEN]; // max 52 byte
} rfmsg;

// Info bits
#define MSG_ENCRYPTED          0x80
#define DEVICE_END           0
#define DEVICE_RANGE_EXTENDER 0x10
#define DEVICE_AP            0x20
#define DEVICE_MASK          0x30
#define HOP_COUNT_MASK       0x0f

// tid
#define RESPONSE_BIT         0x80
#define TID_MASK             0x7f
```

1.2.1 Položka Dst

Dst označuje adresu příjemce zprávy. Její formát je uveden v kapitole „Adresace stanic“. Tato adresa může být buď konkrétní (unicast), nebo skupinová (multicast), nebo všeobecná (broadcast).

1.2.2 Položka Src

Src označuje adresu odesílatele zprávy. Její formát je uveden v kapitole „Adresace stanic“. Tato adresa musí být vždy konkrétní adresou stanice (unicast).

1.2.3 Položka Port

Položka port určuje číslo aplikace, které zprávu zpracovává. Tedy tím určuje i její význam. Vždy se spolu baví aplikace, které rozumí datům, která se posílají.

1.2.4 Položka Info

Položka má délku 1 byte a definuje:

- hop count, což je maximální počet stanic, přes které zpráva prochází. Při průchodu každou stanicí se tato položka zmenší o 1. Pokud dosáhne hodnoty 0, pak je paket zničen. Tato technika zabraňuje nekonečnému kolování paketů v síti.
- device type určuje typ zařízení
- msg encrypted indikuje zašifrovanou zprávu



1.2.5 Položka Tid

Položka má délku 1 byte a definuje:

- tid, číslo transakce. Toto číslo bude buď generované náhodně, nebo se bude zvětšovat o 1 pro každou novou zprávu.
- ack, bit potvrzení, tento bit indikuje potvrzení zprávy s daným tid. V této implementaci nebude použita technika „piggyback“, která umožňuje nová data společně s potvrzením přijatých dat v jedné zprávě.



1.3 Aplikační vrstva

Aplikační vrstva interpretuje datovou část paketu a provádí její zpracování. Konkrétní akce je určena polem *Port* v linkové hlavičce.

Aplikace	Port	Význam
PING	1	Testovací paket
LINK	2	
JOIN	3	
SECURITY	4	
FREQ	5	Testovací pakety, bez odpovědi
MGMT	6	
WOR	7	Probuzení rádiem k aktivitě
HELLO	8	Výzva k ohlášení se modulu
TRACE	9	Trasovací paket, umožňuje sledovat trasu paketu
TEST	31	Testovací pakety, bez odpovědi
VIRTCHAN	32	Virtuální transparentní kanál, přenos nespecifikovaných byte
CONFIG	33	Konfigurační parametry
SISA	34	Vysílání spontánních dat s potvrzením
NEP	35	Dálkové řízení, NEP v jednom paketu
SLIP	36	Dálkové řízení, NEP ve více paketech
SISA_TX	37	Vysílání spontánních dat bez potvrzení

1.3.1 Aplikace PING

Tato aplikace je určena k prověřování dosažitelnosti radiové stanice. Implementace v modulech je taková, že pokud modul dostane tuto zprávu a nemá nastaven bit *ACK*, pak modul tento bit nastaví, a vrátí zprávu na tu adresu, ze které byl tento paket vyslán.

1.3.2 Aplikace VIRTCHAN

Pomocí této aplikace lze několika moduly vytvořit virtuální sběrnici.

1.3.3 Aplikace SISA

Tato aplikace je určena pro spontánní zasílání dat z bateriových modulů. Bateriové moduly pracují v takovém režimu, že radiová část modulu je buď zcela vypnutá, nebo pracuje v režimu WOR. V nastavitelných intervalech pak vysílá data. Data jsou kódována protokolem NEP s typem zprávy DATA/INFO. Moduly očekávají potvrzení zprávy nadřazeným systémem tak, že nadřazený systém pošle SLRF paket zpět stím, že TID je doplněno bitem ACK. Paket může mít nulovou délku nebo může obsahovat proměnné protokolu NEP, které se použijí na nastavení vnitřních proměnných.

1.3.4 Aplikace NEP a SLIP

Tato aplikace se používá na vzdálené řízení modulů pomocí protokolu NEP. Jediný rozdíl mezi NEP a SLIP aplikací je ten, že data NEP zprávy se v protokolu NEP musí vejít do jednoho RF paketu, kdežto u SLIP aplikace mohou být rozložena do více RF paketů s tím, že se použije zapouzdření SLIP protokolem a zabezpečení CRC8 polynomem.

1.3.5 Aplikace WOR

Tato aplikace slouží k probuzení bateriových modulů. Paket zároveň obsahuje informaci, jak se má modul chovat po probuzení. Možnosti probuzení jsou:

- jen do příjmu
- vyslání HELLO paketu
- vynucené vysílání dat modulu

1.3.6 Aplikace HELLO

Tato aplikace slouží typicky jako odpověď na WOR paket, ale může být použita pro jakoukoliv identifikaci radiových modulů. Primárním účelem aplikace je upozornit na existenci radiového modulu. Datová část SLRF paketu je prázdná, důležitá je hlavička, kdy odesílatel o sobě dává vědět, že existuje.

1.4 SLIP protokol v RF paketech

Některé aplikace potřebují pro přenos dat větší pakety, než je 52 byte, které jsou datovou částí samostatných RF paketů. Pro tyto účely bude používán SLIP protokol, do něhož budou datové pakety aplikace zapouzdřeny. Pro kontrolu správnosti přenosu bude použit kontrolní součet CRC8.

1	2	3	4	5	6
Paket aplikace (libovolně dlouhý)					
SLIP zapouzdření					CRC8
RF Paket	RF Paket	RF Paket	RF Paket	RF Paket	

Pro RF pakety bude použito číslování **tid** pro omezení duplicity příchozích paketů vlivem extenderů. Dále je při implementaci nutné dbát na to, že RF pakety různých SLIP paketů mohou najednou přijít z několika stanic. Proto pozor na párování **adres**, **tid**, a aktuálního bufferu právě přijímaného SLIP paketu.

1.5 Adresace systému WACO

WACO protokol používá tyto způsoby adresace:

- monocast address - adresa, identifikující jedno konkrétní zařízení
- multicast address - adresa, identifikující skupinu zařízení (skupinová adresa)
- broadcast address - adresa, identifikující všechna zařízení v dosahu

Adresa je tvořena 4 byte, na které lze rovněž nahlížet jako na jedno 32-bitové číslo. Pokud je první byte 0x00, pak se jedná buď o multicast nebo broadcast adresu. Jinak se jedná o unikátní (monocast) adresu. Adresa je totožná s výrobním číslem zařízení. Broadcast adresa má tvar

0x00 0x00 0x00 0x00

Skupinová adresa má tvar

0x00 0x01 x x

kde poslední 2 byte udávají číslo skupiny.

1.6 Funkce opakovače

Tato funkce může být implementována jak u samostatných zařízení, tak jako součást koncové stanice, implementující aplikační vrstvy protokolu. Používá se záplavové směrování s omezením doby životnosti paketu (hop count) a s náhodnou volbou timeslotu.

Algoritmus je následující:

- pokud stanice přijme paket s její adresou, pak ji zpracuje příslušná aplikační vrstva
- pokud stanice přijme broadcast nebo skupinovou adresu, zpracuje ji její aplikační vrstva a zároveň vytvoří kopii, kterou přepoše
- pokud stanice přijme konkrétní adresu, která jí nepatří, pak ji přepoše
- přeposlání paketu (retransmission) se děje tak, že se dekrementuje položka hop count. Pokud je nulová, paket se zahodí, jinak se zvolí náhodné číslo timeslotu a paket se zařadí do fronty na vysílání.

1.7 Použití bran

Brány do radiové sítě jsou zařízení, která mají radiový modul a další datové rozhraní, kterým může modul komunikovat s okolním světem, Typicky jde o rozhraní USB, Ethernet nebo GSM/GPRS. Nadřazený systém, například PC, komunikuje s bránou protokolem NEP s tím, že brána umožňuje nadřazenému systému posílat a přijímat přímo radiové pakety. Tato vlastnost je dosažitelná tím, že v NEP zprávě se vyskytuje proměnná s OID 119. Obsahem této proměnné jsou všechny byty paketu, který je přijat nebo má být vyslán do radiové sítě.

Kapitola 2

NEP protokol

NEP protokol je komunikační protokol pro výměnu informací v řídicích systémech. Protokol definuje aplikační a síťovou vrstvu pro různá přenosová prostředí. Základní koncepce NEP protokolu vychází ze SNMP protokolu, známého v sítích, založených na protokolu IP, s tím, že NEP protokol musí být snadno implementovatelný do malých mikropočítačů s řádově jednotkami KB paměti programů a stovkami byte paměti RAM.

NEP protokol přebírá základní myšlenku SNMP v tom, že každý komunikační objekt obsahuje seznam tzv. “proměnných”, jejichž obsah je prostřednictvím NEP protokolu zaslán mezi komunikujícími entitami, a to buď:

- na vyžádání (dotaz, odpověď)
- spontánně (poplarchy, spontánní data)

Proměnná může být buď

- skalární, obsahuje jednu hodnotu
- vektor, obsahuje jednorozměrné pole hodnot stejného typu, dosažitelných indexem

Proměnnou definuje následující uspořádaná čtveřice { O, T, L V }, kde

O	znamená „Object ID“, zkráceně OID. Je to celé číslo, které jednoznačně určuje obsah proměnné. V případě proměnné typu vektor obsahuje OID i index.
T	znamená „Type“, určuje typ proměnné, například textový řetězec, nebo číslo
L	znamená „Length“, udává počet datových byte, které proměnná zabírá ve zprávě
V	znamená „Value“, tedy hodnotu proměnné podle typu

Proměnné v systému tvoří strom, takže existuje možnost automatického procházení všech proměnných daného systému, nebo případně určitého podstromu, bez jejich znalosti, obdoba tzv. WALK mechanismu z SNMP protokolu.

Příklady proměnných

Proměnná	OID	Index	Typ	Hodnota
Název zařízení	1	-	0	"SISA modul"
Výrobní číslo	4	-	1	0x11 0x12 0x25
Uptime systému	12	-	2	12322
RF adresa interface	118	1	1	0xff 0xff 0xf9 0x12
Typ zprávy	63	-	2	1

2.1 Definice významu proměnných

Aby jednotlivé komunikující systémy vždy věděly, co která proměnná obsahuje, je firma Softlink s.r.o. jediným autorizovaným subjektem s právem definice nových proměnných. Tento mechanismus slouží k tomu, že při výměně dat mezi různými systémy nemůže dojít k jinému výkladu obsahu proměnné mezi odesilatelem a příjemcem zprávy.

2.2 NEP zpráva

Základní komunikační entitou je NEP zpráva. Jednotlivé systémy komunikují tak, že si přes různá komunikační prostředí vyměňují NEP zprávy.

NEP zpráva je tvořena seznamem proměnných.

Variable1	Variable2	Variable3	...
-----------	-----------	-----------	-----

První proměnná je povinná a definuje typ zprávy.

2.2.1 Typy zpráv

Typ zprávy je definován proměnnou OID 63 (skalární proměnná), typ 2 (celé číslo bez znaménka), jejíž číselná hodnota udává typ zprávy.

NEP protokol rozeznává následující typy zpráv:

- TRAP zasílání alarmů
- SET, nastavování vnitřních proměnných
- GET, žádost o posílání obsahu vnitřních proměnných
- GETNEXT, žádost o poslání obsahu sousední vnitřní proměnné
- RESPONSE, odpověď na GET nebo GETNEXT
- INFO/DATA, asynchronní zpráva bez vyžádání

Použití konkrétního typu zprávy záleží na tom, kdo zprávu generuje. Existují 2 typy zařízení:

- řízená zařízení, ta, od kterých sbíráme data
- řídicí zařízení, ta která sbírají data, nastavují jiná zařízení, případně monitorují jejich činnost

Použití jednotlivých typů zpráv dle typu zařízení udává následující tabulka:

Typ zprávy	Rízené zařízení	Rídící zařízení	Význam
TRAP	odesílá	odesílá/přijímá	Poplachová zpráva
SET	přijímá	odesílá	Nastavení proměnné
GET	přijímá	odesílá	Zjištění hodnoty proměnné
GETNEXT	přijímá	odesílá	Zjištění hodnoty nejbližší "vedlejší" proměnné ve stromu proměnných řízeného zařízení
RESPONSE	odesílá	přijímá	Odeslání hodnoty proměnné
DATA/INFO	odesílá	přijímá	Odeslání hodnoty proměnné

2.2.1.1 Zpráva typu TRAP

Tato zpráva může být generována libovolným typem zařízení. Zpráva dodržuje formát všech NEP zpráv. První proměnnou je tedy typ zprávy (OID=63). Dále je zpráva rozdělena na 2 sekce

- hlavička poplachové zprávy
- tělo poplachové zprávy

Hlavička poplachové zprávy je tvořena proměnnou, která identifikuje odesílatele. Pak následuje libovolný počet proměnných, kromě proměnné s OID=60, což je proměnná TRAP CODE, která udává typ poplachu a tedy i povinný seznam proměnných v těle poplachové zprávy. Tyto proměnné tvoří doplňující informaci, jako je například čas vzniku alarmu.

Tělo poplachové zprávy začíná proměnnou s OID=60, což je TRAP CODE (kód alarmu). Ten udává hlavní informaci o alarmu a určuje interpretaci zbytku zprávy. Za touto proměnnou může následovat žádná nebo několik proměnných libovolného OID, které budeme nazývat závislou proměnnou. Platí však, že pro daný kód TRAP CODE existuje vždy přesně určený seznam proměnných, aby bylo lze vždy jednoznačně zprávu interpretovat.

Příklad poplachové zprávy:

Proměnná	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	0
4	14	-	Cold Start

2.2.1.2 Zprávy typu SET,GET, RESPONSE a DATA/INFO

Tyto typy zpráv mají standardní formát, to znamená, že první proměnnou je typ zprávy (OID=63). Pak následuje seznam libovolných proměnných. Maximální počet těchto proměnných je dán pouze možnostmi síťové vrstvy, která SEP zprávu přenáší. U zprávy typu GET bývá typ proměnné nastaven na typ *Null*. Pokud je ve zprávě RESPONSE u dané proměnné typ *Null*, znamená to, že objekt danou proměnnou nezná, nebo aktuálně nezná její hodnotu.

2.2.1.3 Zpráva typu GETNEXT

Tato zpráva slouží pro sekvenční průchod všemi proměnnými řízeného objektu i v případě, že odesílatel (tazatel) nezná seznam takových proměnných. Podobná situace může nastat i pokud tazatel prochází proměnou typu vektor, a nezná platné indexy (například z důvodů tzv. řídkých vektorů). Zpráva GETNEXT typicky obsahuje pouze povinnou první proměnnou s typem zprávy a poté právě jednu proměnnou označenou ve zprávě typem Null. Tázaný objekt odpoví nejbližší pravou proměnnou ve stromě proměnných.

Pokud chceme dostat z objektu první platnou proměnnou, pošleme SEP zprávu s tímto obsahem.

Proměnná	OID	Index	Typ	Hodnota
1	63	-	2 (Unsigned Int)	2
2	0	-	7 (Null)	

Objekt pak může odpovědět například takto:

Proměnná	OID	Index	Typ	Hodnota
1	63	-	2 (Unsigned Int)	4
2	1	-	1 (String)	"WM868-SJ"

Pokud chce tazatel znát další proměnnou v ve stromu, pošle dotaz:

Proměnná	OID	Index	Typ	Hodnota
1	63	-	2 (Unsigned Int)	2
2	1	-	Null	

a objekt odpoví další proměnnou.

Prohledávání končí, pokud objekt odpoví stejnou proměnnou, jaká byla v dotazu GETNEXT. To znamená, že ve stromu proměnných už žádná další proměnná neexistuje.

2.2.2 Kódování dat aplikační vrstvy

Způsob kódování dat aplikační vrstvy musí umožnit izolování jednotlivých proměnných ve zprávě, zjištění jejich OID (s případným indexem), typu, a hodnoty. Každá proměnná obsahuje jakousi hlavičku, kde je zakódováno OID, index, typ a délka datové části v uvedeném pořadí, za kterou následují data.

Dané kódování dovoluje následující rozsahy:

Položka	Rozsah hodnot
OID	1 ... 16383
Index	1 ... 32767
Type	1 ... 127
Data length	0 ... 32767

2.2.2.1 O Objekt

Objekt určuje konkrétní proměnnou uvnitř systému. Může obsahovat i index, který slouží jako určení řádku v případě, že objekt je sloupec tabulky. Kódování je následující:

D7	D6	D5	D4	D3	D2	D1	D0
LongID	IndexPresent	OID Low/High					
OID Low (optional)							
LongIndex	Index Low/High						
Index Low (optional)							

LongOID určuje, zda číslo objektu je umístěno jen v prvním byte „0“ z intervalu 0 až 63 (6 bitů), nebo je dlouhé z intervalu 0 až 16383 (14 bitů).

IndexPresent „1“ určuje, že objekt je sloupec tabulky a že následuje vlastní index. Ten může mít buď krátkou formu 1 byte, v tom případě je v tomto bytu bit D7=“0“ a index je z intervalu 1 až 127 (index 0 se používá pro speciální případy), nebo je bit D7=“1“ a pak následuje další byte indexu. Celkem index může nabývat hodnot 1 až 32767. U dlouhého tvaru se index složí jako dolních 7 bitů prvního byte indexu jako MSB, a další byte celý jako LSB indexu.

2.2.2.2 T Typ a L Délka

Pro úsporu místa jsou tyto dvě veličiny kódovány pohromadě. Pro toto existují 3 varianty.

Krátká varianta:

V krátké variantě jsou obě veličiny obsaženy v jednom bytu.

7	6	5	4	3	2	1	0
"0"	Type			Length			

Type udává datový typ proměnné, Length délku datové části proměnné v bytech. Rozsah typů v této variantě je 0 až 7, maximální délka dat je 15 bytů.

Dlouhá varianta:

Tato varianta prodlužuje rozsah datových typů na 128 a maximální délku dat na 255 bytů. Její kódování je následující:

7	6	5	4	3	2	1	0
"1"	Type						
"0"	Length						

Maximální varianta:

Tato varianta umožňuje největší rozsah, datový typ je v rozsahu 0 až 127 a délka dat length je 32767. Kódování je následující:

7	6	5	4	3	2	1	0
"1"	Type						
"0"	Length high (bity D14-D8)						
Length low (bity D7-D0)							

Tab. 2.1: Tabulka typů NEP proměnných

0	String text, ASCIIZ řetězec, vhodný pro výpis
1	Bytes, posloupnost bytů, interpretovaná různě
2	Unsigned number, celé číslo bez znaménka MSB... LSB
3	Signed number, celé číslo se znaménkem, dvojkový doplněk, MSB... LSB
4	IP adresa, MSB..LSB
5	Float, reálné číslo, pravděpodobně v notaci IEEE 748 (4 byty)
6	OID, nepromyšlen formát,pro SNMP zprávy
7	Null , nedefinovaná proměnná

2.2.2.3 V hodnota proměnné (value)

Hodnota proměnné je uložena v bytech, následujících za položkou typ/délka dat. Jejich interpretace je dána typem proměnné.

Proměnná typu 0, text

Hodnota takové proměnné je tisknutelný text ve formě ASCIIZ řetězce. Je tedy ukončen bytem s hodnotou 0. Délka proměnné je různá podle délky řetězce.

Proměnná typu 1, posloupnost byte

Hodnota takové proměnné je tvořena posloupností byte, kde SEP protokol nedefinuje význam jednotlivých byte. Toto je zcela na aplikaci. Délka proměnné je různá podle počtu byte.

Proměnná typu 2, celé číslo bez znaménka

Hodnota takové proměnné udává celé číslo bez znaménka. Délka proměnné je různá podle hodnoty čísla. Jednotlivé byty čísla jdou v pořadí MSB až LSB s tím, že vedoucí 0 nemusí být datech proměnné obsaženy. Standardní velikostí čísel je 32 bitový rozsah.

Příklady hodnot a jejich datových částí:

Hodnota	Datová část
200	0xC8
255	0xFF
256	0x01 0x00
300	0x01 0x2C
100000	0x01 0x86 0xA0

Proměnná typu 3, celé číslo se znaménkem

Hodnota takové proměnné udává celé číslo se znaménkem ve dvojkovém doplňku. Délka proměnné je různá podle hodnoty čísla. Jednotlivé byty čísla jdou v pořadí MSB až LSB s tím, že nejvyšší bit MSB bytu v datech zprávu je znaménko. Standardní velikostí čísel je 32 bitový rozsah.

Příklady hodnot a jejich datových částí:

Hodnota	Datová část
200	0xC8
255	0x00 0xFF
-1	0xFF
256	0x01 0x00
300	0x01 0x2C
100000	0x01 0x86 0xA0

Proměnná typu 4, IP adresa

Hodnota této proměnné jsou vždy 4 byte, tak jak jdou byte IP adresy verze 4 za sebou

Příklady hodnot a jejich datových částí:

Hodnota	Datová část
172.16.16.20	0xAC 0x10 0x10 0x14
10.0.0.1	0x0A 0x00 0x00 0x01

Proměnná typu 5, reálné číslo

Hodnota této proměnné jsou vždy 4 byte tak, že vyjadřují přímo reálné číslo podle IEEE 754 ve formátu "single".

Proměnná typu 7, nedefinovaná proměnná

Délka dat tohoto typu proměnné je vždy 0. Tento typ proměnné se používá:

- v dotazech na obsah proměnné, protože nevíme, zda proměnná v dotazovaném systému vůbec existuje, ani neznáme její hodnotu
- v odpovědích na dotaz v případě, že tázaný systém danou proměnnou neobsahuje

2.3 Kódy alarmů

2.3.1 Kód 0, RESET systému

Tento alarm posílá systém v okamžiku startu. Jedinou závislou proměnnou je pak OID=14, kód resetu. Zpráva tedy vypadá například takto:

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	0
4	14	-	0 (Cold Start)

2.3.2 Kód 1, Změna konfigurace

Tento alarm posílá systém v okamžiku změny stavu konfigurace. Jedinou závislou proměnnou je pak OID=15, kód stavu konfigurace. Zpráva tedy vypadá například takto:

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	1
4	15	-	2 (Config OK)

2.3.3 Kód 2, Příliš mnoho alarmů

Tento alarm posílá systém v okamžiku, kdy tabulka alarmů je zaplněna a došlo k dalšímu alarmu. Systém pak změni poslední alarm v tabulce na tento typ alarmu. Jedinou závislou proměnnou je pak OID=59, počet alarmů paměti. Zpráva tedy vypadá například takto:

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	2
4	59	-	20 (počet alarmů)

2.3.4 Kód 10, Příliš nízká teplota

Tento alarm posílá systém v okamžiku, kdy na některé z měřených teplot dojde k poklesu pod nastavený limit. Jedinou závislou proměnnou je pak OID=105/x, což je aktuální hodnota teploty. Zpráva tedy vypadá například takto:

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	10
4	105	2	-20 (-2.0°C)
5	103	2	0 (0°C)

2.3.5 Kód 11, Příliš vysoká teplota

Tento alarm posílá systém v okamžiku, kdy na některé z měřených teplot dojde k nárůstu nad nastavený limit. Jedinou závislou proměnnou je pak OID=105/x, což je aktuální hodnota teploty. Zpráva tedy vypadá například takto:

2.3.6 Kód 12, Teplota v normálu

Tento alarm posílá systém v okamžiku, kdy na některé z měřených teplot dojde k návratu z alarmového stavu do normálu. Jedinou závislou proměnnou je pak OID=105/x, což je aktuální hodnota teploty. Zpráva tedy vypadá například takto:

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	11
4	105	2	630 (63.0°C)
5	103	2	600 (60.0°C)

Variable #	OID	Index	Value
1	63	-	5
2	118	1	FF FF FA D2
3	60	-	12
4	105	2	-20 (-2.0°C)