

Obsah

Úvod.....	2
Obecný popis protokolu Modbus	3
Režimy komunikace	3
Datová jednotka protokolu Modbus (PDU)	3
Modbus kódy funkcí.....	4
Obecný adresní prostor protokolu Modbus	5
Modbus na sériové lince	5
Výchozí nastavení sériové linky	5
Struktura Modbus RTU rámce na sériové lince	5
Výpočet CRC.....	6
Seznam Modbus funkcí.....	6
Modbus adresní prostor přijmače	7
Seznam cívek.....	7
Seznam vstupních registrů.....	7
Seznam uchovávacích registrů.....	8
Popis, syntaxe a příklad použití využívaných Modbus funkcí	9
(0x01) Funkce Čti cívky.....	9
(0x05) Funkce Zapiš jednu cívku	10
(0x0F) Funkce Zapiš více cívek	11
(0x04) Funkce Čti vstupní registry.....	12
(0x03) Funkce Čti uchovávací registry	13
(0x06) Funkce Zapiš jeden uchovávací registr	14
(0x10) Funkce Zapiš více uchovávacích registrů	15



Úvod

Tento dokument slouží k popisu Modbus protokolu použitého u čidel přijmačů řady NL-MRF-RX s komunikačními rozhraními RS485. Verze tohoto návodu je určena **pro čidla s verzí Firmware 109 a výše**. Číslo verze FW je uvedeno na samolepícím štítku umístěném na plošném spoji uvnitř čidla

Na úvod uvedme několik užitečných informací, k řešení případných potíží:

Z přijmače je možné vyčítat pouze ty registry, které jsou na přijmači dostupné. V opačném případě čidlo reaguje chybovou odpovědí s kódem chyby 0x02 – Ilegální adresa dat.

V případě, že přijmač nekomunikuje, ujistěte se, že Vámi odesílané rámce jsou správné a zkontrolujte, zda na komunikační sběrnici dodržujete pauzy minimálně 4ms, pro správnou detekci konců rámce.

Přijmač pracuje v režimu tzv. **Half-duplex**. To znamená, že není schopno přijímat další požadavky do doby, než odpoví na předchozí modbus rámeček.

Pro případnou **kontrolu** či ověření správnosti **výpočtů modbus crc** je možné využít on-line kalkulátoru:

<https://www.lammertbies.nl/comm/info/crc-calculation.html>

Kalkulátor je nutné přepnout na zadávání HEX znaků a následná výsledek CRC-16 (Modbus) má však v modbus rámci zaměněný horní a dolní byte.



Obecný popis protokolu Modbus

Modbus protokol je Master-Slave protokol. Na sběrnici je přítomen pouze 1 master a až 247 slave zařízení (v našem případě čidel). Komunikaci vždy iniciuje master zařízení. Slave pouze odpovídá na požadavky master zařízení. Modbus využívá Big-endian reprezentaci dat. To znamená že u položek větších jak 1 B je nejdříve je odeslán nejvyšší byte a nejnižší byte až na konec.

Režimy komunikace

Unicast režim:

Master osloví **jedno konkrétní slave zařízení** pomocí jeho Modbus adresy. Slave zprávu zpracuje a odpoví.

Broadcast režim:

Master osloví **všechna slave zařízení** pomocí **broadcast adresy 0**. Všechna slave zařízení zprávu zpracují, ale žádné na ni neodpoví. Broadcast požadavky master zařízení musejí být nutně **příkazy k zápisu**.

Datová jednotka protokolu Modbus (PDU)

Modbus funkce	Data
1 B	N* 1 B

Protokol Modbus definuje tři základní typy PDU:

- 1) **Request PDU** - Slouží k oslovení jednoho či více slave zařízení masterem.

Pole Modbus funkce obsahuje kód dané Modbus funkce. Pole data pak dle dané Modbus funkce adresy, počet proměnných, hodnoty proměnných a jiné

- 2) **Response PDU** - Slouží k odeslání **kladné odpovědi** slave zařízením na přijatou Request PDU.

Pole **Modbus funkce** obsahuje **stejnou hodnotu** jako v přijatém Request PDU. Datová část pak dle dané Modbus funkce provozní hodnoty, přečtené vstupy, cívky ...

- 3) **Exception Response PDU** - Slouží k odeslání **záporné odpovědi** slave zařízením na přijatou Request PDU.

Pole **Modbus funkce** obsahuje **hodnotu Modbus funkce z Request PDU + 0x80** jako signalizace neúspěchu. Datová část pak **identifikuje chybu**.



Chybové kódy v Exception Response PDU

Kód	Druh kódu funkce	Význam
0x01	Ilegální Modbus funkce	Požadovaná Modbus funkce není serverem (čidlem) podporována
0x02	Ilegální adresa dat	Zadaná adresa (cívky, registru ...) je mimo serverem podporovaný rozsah
0x03	Ilegální hodnota dat	Předávaná data jsou neplatná
0x04	Selhání zařízení	Při provádění požadavku došlo k neodstranitelné chybě
0x05	Potvrzení	Kód určený k použití při programování. Server hlásí přijetí platného požadavku, ale jeho vykonání bude trvat delší dobu
0x06	Zařízení je zaneprázdněné	Kód určený k použití při programování. Server je zaneprázdněn vykonáváním dlouho trvajících příkazů.
0x08	Chyba parity paměti	Kód určený k použití při práci se soubory. Server při pokusu přečíst soubor zjistil chybu parity
0x0A	Brána – přenosová cesta nedostupná	Kód určený k práci s bránou (gateway). Brána není schopná vyhradit interní přenosovou cestu od vstupního portu k výstupnímu. Pravděpodobně je přetížená nebo nesprávně nastavená.
0x0B	Brána – cílové zařízení neodpovídá	Kód určený k práci s bránou (gateway). Cílové zařízení neodpovídá, pravděpodobně není přítomno.

Modbus kódy funkcí

- 1) Veřejné kódy funkcí** - Jsou jasně definovány a veřejně zdokumentovány. Je zaručena jejich unikátnost. Obsahují i některé nevyužité kódy pro budoucí využití.
- 2) Uživatelsky definované kódy funkcí** - Umožňují uživateli implementovat funkci, která není protokolem definována. Není garantována unikátnost kódu.

Rozsahy kódů Modbus funkcí

Kód funkce	Druh kódu funkce
1 ... 64	Veřejné kódy funkcí
65 ... 72	Uživatelsky definované kódy funkcí
73 ... 100	Veřejné kódy funkcí
101 ... 110	Uživatelsky definované kódy funkcí
111 ... 127	Veřejné kódy funkcí



Obecný adresní prostor protokolu Modbus

Adresní prostor protokolu Modbus je založen na sadě tabulek s charakteristickými významy. Definovány jsou tyto čtyři základní tabulky:

Tabulka	Popis	Přístup	Adresní prostor (není podmínkou)
Diskrétní vstupy	1-bit	Pouze čtení	0x2710 až 0x4E1F
Cívky	1-bit	Čtení i zápis	0x0000 až 0x270F
Vstupní registry	16-bitů	Pouze čtení	0x7530 až 0x9C3F
Uchovávací registry	16-bitů	Čtení i zápis	0x9C40 až 0xC34F

Modbus na sériové lince

Výchozí nastavení sériové linky

Protokol Modbus RTU definuje výchozí nastavení sériové linky následovně:

Baud rate 19200

1 start bit

8 datových bitů

1 sudý paritní bit

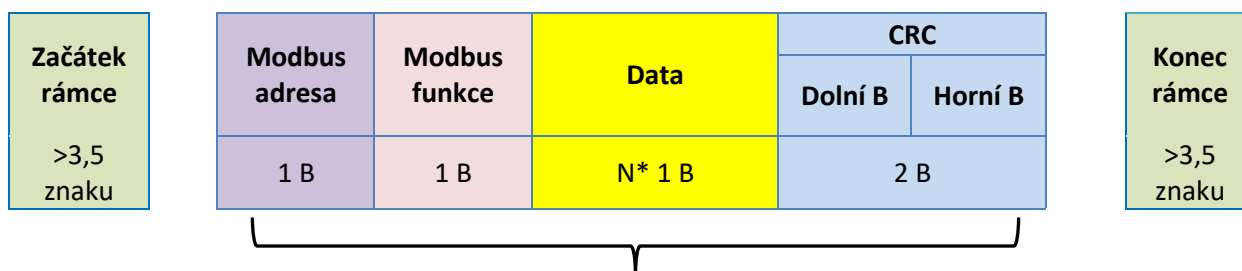
1 stop bit

Tomuto nastavení odpovídají výchozí hodnoty cívek "Modbus parita" a "Modbus stop bit" a uchovávacího registru "Modbus baud rate".

Dále je doporučeno implementovat Baud rate 9600 a možnost žádné parity, přičemž je tento paritní bit nahrazen druhým stop bitem. Možnost dvou stop bitů není podporována.

V režimu Modbus RTU je 1 B složen ze dvou čtyřbitových hexa znaků. Vysílání Modbus rámce je započato a ukončeno pomlčkou na sběrnici delší jak 3,5 znaku. Během vysílání rámce nesmí být mezery mezi jednotlivými znaky větší jak 1,5 znaku.

Struktura Modbus RTU rámce na sériové lince



Modbus rámec na sériové lince
Maximální délka Modbus RTU rámce omezena na 32 B.



Adresování slave zařízení:

<i>adresa</i>	<i>význam</i>
0	Broadcast adresa
1 až 247	Individuální slave adresy
248 až 255	Rezervováno

V Modbus RTU rámcích nesoucích odpovědi určené pro master zařízení je ponechána Modbus adresa odpovídajícího slave zařízení.

Výpočet CRC

Výpočet CRC je prováděn z celého rámce včetně Modbus slave adresy, modbus funkce a datové části rámce

1. Inicializace 16-bit CRC registru na 0xFFFF.
2. Provedeme XOR prvních 8 bitů rámce se spodním bytem CRC registru a výsledek uložíme do CRC registru.
3. Provedeme posun CRC registru o 1 b doprava (směrem k LSB), MSB CRC registru vyplníme 0. Zachytíme a vyhodnotíme nejspodnější bit, který posunem vypadl.
4. Pokud byl tento bit roven 1, provedeme XOR mezi CRC registrem a hodnotou 0xA001 (generující polynom= $1+x^2+x^{15}+x^{16}$). Výsledek uložíme opět do CRC registru.
5. Opakujeme kroky 3 a 4 dokud nebude provedeno osm posunů CRC registru.
6. Provedeme XOR dalších 8 bitů rámce se spodním bytem CRC registru a opakujeme kroky 3 až 5.
7. Takto pokračujeme až do posledního byte rámce.
8. Výsledná hodnota CRC výpočtu je uložena v CRC registru.
9. Při umístění CRC hodnoty do Modbus rámce je nutné zaměnit horní a spodní byte CRC registru (viz struktura Modbus RTU rámce na sériové lince).

Seznam Modbus funkcí

<i>Určení</i>	<i>Kód funkce</i>	<i>Příkaz</i>	<i>Veř./Uživ.</i>
Cívky	0x01	Čti cívky	Veřejná
	0x05	Zapiš jednu cívku	Veřejná
	0x0F	Zapiš více cívek	Veřejná
Vst. registry	0x04	Čti vstupní registry	Veřejná
Uch. registry	0x03	Čti uchovávací registry	Veřejná
	0x06	Zapiš jeden uchovávací registr	Veřejná
	0x10	Zapiš více uchovávacích registrů	Veřejná
	⋮	⋮	⋮

* Jiné Modbus funkce nejsou podporovány



Modbus adresní prostor přijímače

Seznam cívek

<i>cívka</i>	<i>Adresa cívky</i>	<i>Popis</i>	<i>Výchozí hodnota</i>	<i>R/W</i>
Modbus parita	0x0000	Parita 0 = sudá, 1 = žádná	0	R/W
:	:	:	:	:

Seznam vstupních registrů

<i>Vst. registr</i>	<i>Adresa registru</i>	<i>Popis</i>	<i>Jednotky</i>	<i>Formát</i>	<i>R/W</i>
Stav přijímače	0x7530	Stav přijímače		uint16	R
		bit 0 - signalizace bezchybného provozu přijímače	0-chyba / 1-ok		
		bit 1 - signalizace sepnutého Relé 1	0-off / 1-on		
		bit 2 - signalizace sepnutého Relé 2	0-off / 1-on		
		bit 3 – ztráta spojení čidla 1	0-ne / 1-ano		
		bit 4 – ztráta spojení čidla 2	0-ne / 1-ano		
		bit 5 – ztráta spojení čidla 3	0-ne / 1-ano		
		bit 6 – ztráta spojení čidla 4	0-ne / 1-ano		
		bit 7 – ztráta spojení čidla 5	0-ne / 1-ano		
		bit 8 – ztráta spojení čidla 6	0-ne / 1-ano		
		bit 9 – ztráta spojení čidla 7	0-ne / 1-ano		
		bit 10 – ztráta spojení čidla 8	0-ne / 1-ano		
		bit 11 – ztráta spojení čidla 9	0-ne / 1-ano		
		bit 12 – ztráta spojení čidla 10	0-ne / 1-ano		
	bit 13-15 - nevyužito				
Čidlo 1 - CO2	0x7531	Čidlo 1 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 1 - RH	0x7532	Čidlo 1 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 1 - T	0x7533	Čidlo 1 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 2 - CO2	0x7534	Čidlo 2 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 2 - RH	0x7535	Čidlo 2 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 2 - T	0x7536	Čidlo 2 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 3 - CO2	0x7537	Čidlo 3 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 3 - RH	0x7538	Čidlo 3 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 3 - T	0x7539	Čidlo 3 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 4 - CO2	0x753A	Čidlo 4 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 4 - RH	0x753B	Čidlo 4 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 4 - T	0x753C	Čidlo 4 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 5 - CO2	0x753D	Čidlo 5 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 5 - RH	0x753E	Čidlo 5 - Relativní vlhkost vzduchu	%	uint16	R



Seznam vstupních registrů

Vst. registr	Adresa registru	Popis	Jednotky	Formát	R/W
Čidlo 5 - T	0x753F	Čidlo 5 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 6 - CO2	0x7540	Čidlo 6 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 6 - RH	0x7541	Čidlo 6 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 6 - T	0x7542	Čidlo 6 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 7 - CO2	0x7543	Čidlo 7 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 7 - RH	0x7544	Čidlo 7 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 7 - T	0x7545	Čidlo 7 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 8 - CO2	0x7546	Čidlo 8 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 8 - RH	0x7547	Čidlo 8 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 8 - T	0x7548	Čidlo 8 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 9 - CO2	0x7549	Čidlo 9 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 9 - RH	0x754A	Čidlo 9 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 9 - T	0x754B	Čidlo 9 - Teplota okolního vzduchu	0,1 °C	uint16	R
Čidlo 10 - CO2	0x754C	Čidlo 10 - Koncentrace oxidu uhličitého	ppm	uint16	R
Čidlo 10 - RH	0x754D	Čidlo 10 - Relativní vlhkost vzduchu	%	uint16	R
Čidlo 10 - T	0x754E	Čidlo 10 - Teplota okolního vzduchu	0,1 °C	uint16	R
⋮	⋮	⋮	⋮	⋮	⋮

Pokud čidlo nějakou veličinu neobsahuje (neumí ji měřit), tak základna odešle hodnotu 9999. Například pokud se snažíme vyčíst z prvního napárovaného čidla CO₂, RH a T (tabulka Seznam vstupních registrů, adresy 0x7531(CO₂) až 0x75339(T)) ale první napárované čidlo je např. čidlo NLB-RH+T, tak místo platné hodnoty CO₂ vyčte hodnotu 9999. Stejně tomu je u RH a teploty (například pokud by bylo napárované pouze čidlo CO₂ tak místo RH a T by bylo 9999).

Seznam uchovávacích registrů

Uch. Registr	Adresa registru	Popis	rozsah	Výchozí hodnota	R/W
Modbus adresa	0x9C40	Modbus slave adresa čidla	1 až 247	1	R/W
Modbus baud rate	0x9C41	komunikační tok sběrnice	¹⁾	19200	R/W
⋮	⋮	⋮	⋮	⋮	⋮

¹⁾ Podporovány jsou pouze následující komunikační rychlosti 4800, 9600, 19200



Popis, syntaxe a příklad použití využívaných Modbus funkcí

(0x01) Funkce Čti cívky

Funkce slouží ke čtení stavu po sobě následujících cívek. V požadavku je specifikována adresa první vyčítané cívky a počet cívek k vyčtení. V odpovědi je pak délka stavů cívek a stavy jednotlivých cívek přičemž jsou odesílány po 8 cívkách vždy od MSB k LSB. Je-li některý bajt neúplný, je doplněn nulami na pozicích od MSB směrem k LSB. **Log 1 = ON, Log 0 = OFF**

1) Request PDU

Modbus funkce 0x01 1 B	Adresa 1. cívky viz Seznam cívek 2 B	Počet cívek 2 B
-------------------------------------	---	---------------------------

Příklad vyčtení cívek Modbus parita:

0x01	Modbus parita 0x00 0x00	Počet cívek =1 0x00 0x01
------	-----------------------------------	------------------------------------

2) Response PDU

Modbus funkce 0x01 1 B	Počet bytů N 1 B	Stavy cívek N B
-------------------------------------	-------------------------------	---------------------------

$N = \text{Počet cívek}/8$, je-li zbytek po dělení nenulový pak $N = (\text{Počet cívek}/8)+1$

Příklad odpovědi pro parita žádná:

0x01	Počet bytů 0x01	Stavy cívek =1 0x01
------	---------------------------	-------------------------------

3) Exception Response PDU

Modbus funkce 0x80 0x81 1 B	Chybový kód 1,2,3 nebo 4 1 B
--	---



(0x05) Funkce Zapiš jednu cívku

Tato funkce slouží k nastavení jedné cívky do stavu ON nebo OFF. **0x0000 = OFF, 0xFF00 = ON**. Pozitivní odpověď je kopií požadavku.

1) Request PDU

Modbus funkce	Adresa cívky	Stav cívky
0x05	viz Seznam cívek	0x0000 nebo 0xFF00
1 B	2 B	2 B

Příklad zapsání cívky Modbus parita na žádná:

Modbus funkce	Modbus Parita	Stav cívky = ON
0x05	0x00 0x00	0xFF 0x00

2) Response PDU

Modbus funkce	Adresa cívky	Stav cívky
0x05	viz Request PDU	viz Request PDU
1 B	2 B	2 B

Příklad odpovědi na zapsání cívky Modbus parita na žádná:

Modbus funkce	Modbus Parita	Stav cívky = ON
0x05	0x00 0x00	0xFF 0x00

3) Exception Response PDU

Modbus funkce 0x80	Chybový kód
0x85	1,2,3 nebo 4
1 B	1 B



(0x0F) Funkce Zapiš více cívek

Tato funkce louží k zapsání po sobě jdoucích cívek. V požadavku je specifikována adresa první zapisované cívky, počet cívek, délka stavů cívek a stavy cívek přičemž jsou odesílány po 8 cívkách vždy od MSB k LSB. Je-li některý bajt neúplný, je doplněn nulami na pozicích od MSB směrem k LSB. **Log 1 = ON, Log 0 = OFF**. V odpovědi je pak adresa první cívky a počet zapsaných cívek.

1) Request PDU

Modbus funkce 0x0F 1 B	Adresa 1. cívky viz Seznam cívek 2 B	Počet cívek 2 B	Počet bytů N 1 B	Stavy cívek N B
-------------------------------------	---	---------------------------	-------------------------------	---------------------------

N = Počet cívek/8, je-li zbytek po dělení nenulový pak N= (Počet cívek/8)+1

Příklad zapsání cívek Modbus parita OFF:

0x0F	Modbus parita 0x00 0x00	Počet cívek = 1 0x00 0x01	Počet bytů = 1 0x01	Stavy cívek 0x01
------	--------------------------------------	--	-------------------------------	----------------------------

2) Response PDU

Modbus funkce 0x0F 1 B	Adresa 1. cívky viz Request PDU 2 B	Počet cívek viz Request PDU 2 B
-------------------------------------	--	--

Příklad odpovědi na zapsání cívek Modbus parita OFF:

0x0F	Modbus parita 0x00 0x00	Počet cívek = 1 0x00 0x01
------	--------------------------------------	--

3) Exception Response PDU

Modbus funkce 0x80 0x8F 1 B	Chybový kód 1,2,3 nebo 4 1 B
--	---



(0x04) Funkce Čti vstupní registry

Tato funkce slouží ke čtení obsahu souvislého bloku vstupních registrů. V požadavku je specifikována adresa prvního registru a počet registrů. V odpovědi odpovídá každému registru dvojice bytů.

1) Request PDU

Modbus funkce	Adresa 1. registru viz Seznam vst. reg.	Počet registrů
0x04		1 až max 13
1 B	2 B	2 B

Příklad vyčtení vstupních registrů Čidlo 1 - CO2 a Čidlo 1 - RH:

0x04	Čidlo 1 - CO2 0x75 0x31	Počet registrů = 2 0x00 0x02

2) Response PDU

Modbus funkce	Počet bytů	Stavy registrů
0x04	2*N	
1 B	1 B	2*N B

N = Počet registrů (viz Request PDU)

Příklad odpovědi na vyčtení vstupních registrů Čidlo 1 - CO2 a Čidlo 1 - RH:

0x04	Počet bytů 0x04	CO2 = 980 ppm 0x03 0xD4	RH = 335 ‰ 0x01 0x4F

3) Exception Response PDU

Modbus funkce 0x80	Chybový kód
0x84	1,2,3 nebo 4
1 B	1 B



(0x03) Funkce Čti uchovávací registry

Tato funkce slouží ke čtení obsahu souvislého bloku uchovávacích registrů. V požadavku je specifikována adresa prvního registru a počet registrů. V odpovědi odpovídá každému registru dvojice bytů.

1) Request PDU

Modbus funkce	Adresa 1. registru viz Seznam uch. reg.	Počet registrů
0x03	2 B	1 až max 13
1 B		2 B

Příklad vyčtení uchovávacích registrů Modbus adresa a modbus baud rate:

0x03	Modbus adresa	Počet registrů = 2
	0x9C 0x40	0x00 0x02

2) Response PDU

Modbus funkce	Počet bytů	Stavy registrů
0x03	2*N	
1 B	1 B	2*N B

N = Počet registrů (viz Request PDU)

0x03	Počet bytů	MB adresa = 1	MB baud rate = 19200
	0x06	0x00 0x01	0x4B 0x00

3) Exception Response PDU

Modbus funkce 0x80	Chybový kód
0x83	1,2,3 nebo 4
1 B	1 B



(0x06) Funkce Zapiš jeden uchovávací registr

Tato funkce slouží k zápisu jednoho uchovávacího registru. V požadavku je specifikována adresa registru, který se má zapsat a hodnota, která se má zapsat. Normální odpověď je kopií požadavku a je vrácena poté, co je registr zapsán.

1) Request PDU

Modbus funkce	Adresa registru viz Seznam uch. reg.	Stav registru
0x06		0x0000 až 0xFFFF
1 B	2 B	2 B

Příklad zapsání uchovávacího registru Modbus adresa na 10:

0x06	Modbus adresa	Stav registru = 10
	0x9C 0x40	0x00 0x0A

2) Response PDU

Modbus funkce	Adresa registru viz Request PDU	Stav registru viz Request PDU
0x06		
1 B	2 B	2 B

Příklad odpovědi na zapsání uchovávacího registru Modbus adresa na 10:

0x06	Modbus adresa	Stav registru = 10
	0x9C 0x40	0x00 0x0A

3) Exception Response PDU

Modbus funkce 0x80	Chybový kód
0x86	1,2,3 nebo 4
1 B	1 B



(0x10) Funkce Zapiš více uchovávacích registrů

Tato funkce slouží k zápisu souvislého bloku uchovávacích registrů. V požadavku je specifikována adresa prvního registru, který se má zapsat, počet registrů a hodnoty, které se mají zapsat. Normální odpověď obsahuje počáteční adresu a počet zapsaných registrů.

1) Request PDU

Modbus funkce	Adresa 1. registru viz Seznam uch. reg.	Počet registrů	Počet bytů	Stavy registrů
0x10		1 až max 11	2*N	
1 B	2 B	2 B	1 B	2*N B

N = Počet registrů

Příklad zapsání uchovávacích registrů Modbus adresa a modbus baud rate:

0x10	Modbus adresa	Počet registrů = 2	Počet bytů = 4	Mb adr = 10	MB baud rate = 9600
	0x9C 0x40	0x00 0x02	0x04	0x00 0x0A	0x25 0x80

2) Response PDU

Modbus funkce	Adresa 1. registru viz Request PDU	Počet registrů viz Request PDU
0x10		
1 B	2 B	2 B

N = Počet registrů

Příklad odpovědi na zapsání uchovávacích registrů Modbus adresa a modbus baud rate:

0x10	Modbus adresa	Počet registrů = 2
	0x9C 0x40	0x00 0x02

3) Exception Response PDU

Modbus funkce 0x80	Chybový kód
0x90	1,2,3 nebo 4
1 B	1 B

