RACOM
RADIO DATA NETWORKS

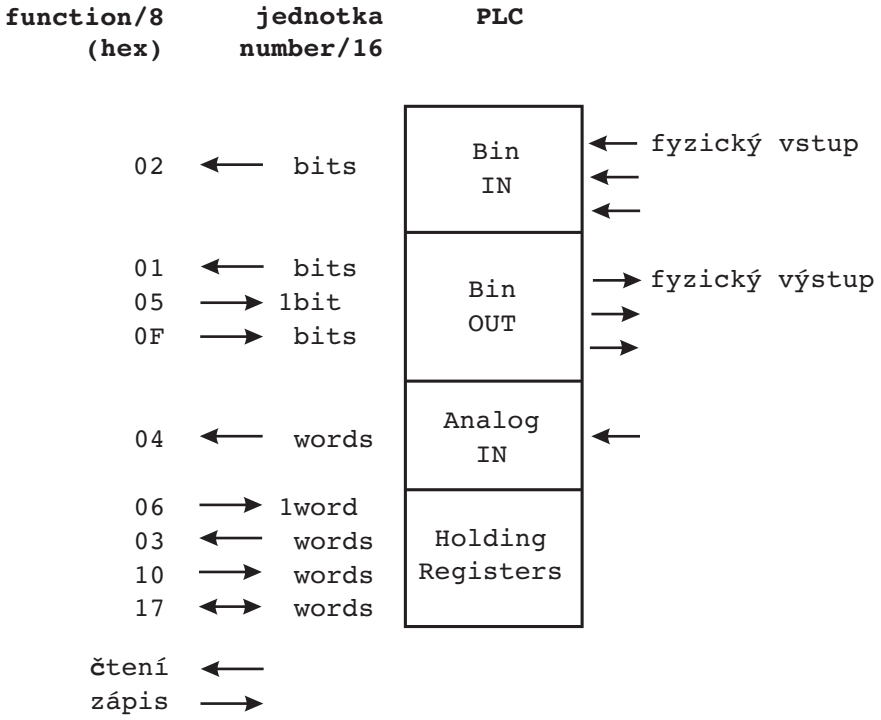
Formát rámce MODBUS pro MORSE

verze x.xx
12. ledna 2011

1. Úvod

Modbus je typický představitel rodiny protokolů určených pro sběrnici realizovanou na RS485. Používá 256bajtové rámce opatřené 16bitovým CRC. Protože Modbus rozlišuje typy přenášených dat (bitové, bajtové, wordové), jsou zavedny typy rámců pro odlišení těchto variant. Modbus typ rámce popisuje číslem funkce, kterou rámec realizuje. Dále je uveden přehled nejčastějších funkcí protokolu Modbus.

2. Přehled funkcí Modbusu pro čtení a zápis z různých částí paměti PLC:



3. Popis jednotlivých funkcí:

3.1. - 01 - (Read Output Status)

Čte z automatu Slave stavy vystupů (relé, tranzistorové spínače, apod.). Zakladním jednotkou čtenou pomocí této funkce je jeden výstup - v našich poměrech je to jeden bit. Protože ovšem protokol umí

přenést jako nejmenší element jeden bajt, jsou bity výstupů sdružovány do bajtů. Master může adresovat více bitů naráz, pak je obdrží naskládané do několika bajtů odpovědi.

Typické tvary rámců pak vypadají takto:

výzva

```
| adr/8 | fce/8 | start/16 | number/16 | crc/16 |
```

odpověď

```
| adr/8 | fce/8 | cnt/8 | data/8 * cnt | crc/16 |
```

adr - adresa automatu na sběrnici Modbus, tato adresa musí být jedinečná v rámci celé sběrnice. Adresa je stejná jak u dotazu tak i odpovědi.

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat (výstupu), která budou zpracována

number - počet položek (bitů), které budou zpracovány

cnt - počet datových bajtů rámce

data - vlastní data dotazu zarovnaná na osm bitů

adr - zabezpečovací slovo

Příklad:

Mějme stav výstupů v našem automatu od nulté adresy 0x1480. Automat má adresu na Modbusu 0x10. Pomocí funkce 01 nejprve přečteme všechny naráz a pak jen druhou polovinu.

Dotaz pro první případ:
`1001 0000 0010 crc` – chceme 16 výstupů od nulté pozice

odpověď:
`1001 0214 80 crc` – vráceno všech 16 výstupů. Je potřeba upozornit, že data nejsou zarovnávána na sudý počet bajtů, jak je obvyklé v MORSE sítích.

Dotaz pro druhý případ:
`1001 0008 0008 crc` – chceme 8 výstupů od osmé pozice

odpověď:

`1001 0180 crc` – vráceno 8 výstupů. Jak je vidno, výstupy jsou organizovány ve formátu Intel (malý endian).

3.2. - 02 - (Read Input Status)

Tato funkce je naprosto totožná s předcházející funkcí, ale čte ze Slave automatu vstupy. Tvary rámců jsou rovněž shodné.

3.3. - 03- (Read Holding Registers)

Funkce vrací z automatu 16bitové paměťové registry. Tyto registry jsou obecně použitelné (General purpose). V našem případě jsou použité pro cache a paketový režim. Rámce pak vypadají takto:

výzva

```
| adr/8 | fce/8 | start/16 | number/16 | crc/16 |
```

odpověď

```
| adr/8 | fce/8 | cnt/8 | data/8 * cnt | crc/16 |
```

adr - adresa automatu na sběrnici Modbus

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat (výstupu), která budou zpracována

number - počet položek (wordy), které budou zpracovány

cnt - počet datových bajtů rámce

data - obsah požadovaných registrů zarovnaný na 16 bitů

crc - zabezpečovací slovo

Příklad:

V našem automatu z přecházejícího příkladu jsou obsahy tří registů od nulté adresy `0x1480, 0x3450` a `0x4054`.

Dotaz:
`1003 0000 0003 crc`

Odpověď:
`1003 0614 8034 5040 54 crc` – opět rámec není zarovnán na sudý počet bajtů.

3.4. - 04- (Read Input Registers)

Funkce je v principu shodná s funkcí pro čtení registrů. Na rozdíl od ní ovšem vrací stavy analogových vstupů.

3.5. - 05- (Force Single Output)

Tato funkce nastavuje jeden výstup, t.j. jeden bit. Protože binární výstup lze jen nastavit nebo smazat, jsou tvary povely velmi jednoduché.

výzva
`| adr/8 | fce/8 | start/16 | 0xFF00 | crc/16 |` – pro nastavení 1 výstupu
`| adr/8 | fce/8 | start/16 | 0x0000 | crc/16 |` – pro smazání 1 výstupu

odpověď
`| adr/8 | fce/8 | start/16 | 0xFF00 | crc/16 |` nebo

`| adr/8 | fce/8 | start/16 | 0x0000 | crc/16 |` – je prostou kopií dotazu.

adr - adresa automatu na sběrnici Modbus

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat (výstupu), která budou zpracována

crc - zabezpečovací slovo

3.6. - 06- (Preset Single Register)

Nastaví obsah jednoho registru = 1 word. Je podobná funkci předcházející, jen místo stavu bitu se objevuje stav registru.

výzva
`| adr/8 | fce/8 | start/16 | data/16 | crc/16 |`

odpověď
`| adr/8 | fce/8 | start/16 | data/16 | crc/16 |` – je opět prostou kopií dotazu

adr - adresa automatu na sběrnici Modbus

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat, která budou zpracována

data - obsah zapisovaného registru

crc - zabezpečovací slovo

3.7. - 0F hex- (Force Multiple Outputs)

Současné nastavení více výstupů.

výzva
`| adr/8 | fce/8 | start/16 | number/16 | cnt/8 | data/8 * cnt | crc/16 |`

odpověď
`| adr/8 | fce/8 | start/16 | number/16 | crc/16 |`

adr - adresa automatu na sběrnici Modbus

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat (výstupu), která budou zpracována

number - počet bitů pro zápis

cnt počet byte nutných pro přenos adresované skupiny bitů

data - stavy zapisovaných výstupů, wordy obsahují prohozené bajty ve tvaru L,H, L,H, L,H, ...

crc - zabezpečovací slovo

3.8. - `10 hex-` (Preset Multiple Regs)

Podobně jako předcházející funkce nastavuje více registrů současně.

výzva
```
| adr/8 | fce/8 | start/16 | number/16 | cnt/8 | data/16 * numb | crc/16 |
```

odpověď
```
| adr/8 | fce/8 | start/16 | number/16 | crc/16 |
```

adr - adresa automatu na sběrnici Modbus

fce - funkce, kterou automat vykoná po přijetí rámce

start - počáteční adresa dat (výstupu), která budou zpracována

number - počet wordů pro zápis

cnt - počet byte nutných pro přenos požadované skupiny wordů

data - stavy zapisovaných registrů

crc - zabezpečovací slovo

3.9. – `17 hex-` (READ/WRITE HOLDING REGISTERS)

Spojuje funkci `03` čtení a `06` zápis.

výzva
```
| a/8 | f/8 | rst/16 | rno/16 | wst/16 | wno/16 | wcnt/8 | wdata/cnt*8
|crc/16 |
```

odpověď
```
| a/8 | f/8 | cnt/8 | data/cnt*8 | crc/16 |
```

a - adresa slave

f - funkce READ/WRITE HOLDING REGISTERS

rst - start čtené oblasti

rno - počet registrů čtené oblasti

wst - start zapisované oblasti

wno - počet registrů zapisované oblasti

wcnt - počet byte dat v rámci

wdata - data, která mají být zapsána

Pokud Slave automat čemukoliv v dotazu/povelu nerozumí je povinnen vrátit výjimku (exception). Výjimka by měla informovat Mastera o situaci, ve které se pokusil pracovat buď nedovolenou fukncí pro daný Slave nebo s daty mimo platný rozsah pro daný Slave automat.

- odpověď na dotaz nebo na povel s chybným zadáním

```
| adr/8 | 0x80+fce /8 | excode/8 | crc/16 |
```

adr - adresa automatu na sběrnici Modbus

fce - funkce, která výjimku vyvolala

excode - číslo výjimky, specifikuje kde přesně a jaká se stala chyba

- 1 - chybné číslo funkce
- 2 - chybná adresa dat
- 3 - chybný obsah dat
- 5 - potvrzené přijetí povelu, jehož provedení je pomalé
- 6 - odmítnutí, Slave je zaměstnán výkonem pomalého povelu

crc - zabezpečovací slovo