

Ovládání modulu PIC08

Popis formátu příkazů

Příručka uživatele a programátora



SofCon[®] s.r.o.
Střešovická 49
162 00 Praha 6
tel/fax: (02) 20 180 454
E-mail: sofcon@sofcon.cz
www : <http://www.sofcon.cz>

Obsah :

1. Úvod	1
2. Soubor příkazů modulů PIC	1
2.1. Status procesoru	1
2.2. EEPROM konfigurace	1
2.3. Adresa uzlu	1
2.4. Čtení vstupního portu IN0, IN1	2
2.5. Zápis na výstupní port OUT0, OUT1	2
2.6. Nastavení měřicího vstupu čítače 0	2
2.7. Předdělič pro čítač 0	2
2.8. Start měření čítače 0	2
2.9. Čtení hodnoty čítače 0	3
2.10. Nastavení měřicího vstupu čítače 1	3
2.11. Předdělič pro čítač 1	3
2.12. Start měření čítače 1	3
2.13. Čtení hodnoty čítače 1	3
3. Konfigurace po startu	4
4. Komunikační protokol	4
4.1. Požadavek na zápis	4
4.2. Požadavek čtení	5

1. Úvod

S moduly PIC se komunikuje přes sériové rozhraní RS485. Moduly po této komunikaci přijímají příkazy a předávají data. Moduly jsou vybaveny programovým vybavením se standardním uživatelským interfacem k tomuto rozhraní, který odpovídá protokolu LECOM. Díky tomu je pro všechny moduly stejný komunikační protokol a podobná struktura a význam příkazů. Pokud jde o stejné příkazy různých modulů, mají stejný kód.

2. Soubor příkazů modulů PIC

Příkaz je plně identifikován kódem příkazu. Kód příkazu je v tomto dokumentu uváděn dekadicky. Za kódem mohou následovat data. Počet byte dat je vyjádřen počtem A u popisu příkazu. Směr přenosu R - Read odpovídá požadavku na čtení podle protokolu Lecom, W - Write odpovídá požadavku na zápis podle protokolu Lecom. V datové položce W značí typ word, R značí typ Real a S značí typ string. Počet písmen vyjadřuje počet hodnot daného typu v datové položce.

2.1. Status procesoru

kód příkazu	data	směr přenosu
0	W	R/W

Čtení, stavového slova procesoru.
Zápis stavového slova procesoru.

Bit 7 indikuje výpadek sítě. Bit 0 a 1 ovládá způsob nastavení obsahu portů procesoru po zapnutí napájení. Buď se port nastaví obsahem z EEPROM nebo se nastaví na log. 0 či na log. 1. Význam obsahu EEPROM je popsán níže.

2.2. EEPROM konfigurace

kód příkazu	data	směr přenosu
1	W	R/W

Čtení obnoví konfiguraci modulu podle obsahu EEPROM.
Zápis uloží konfiguraci modulu do EEPROM.

Přijatá data jsou inorována. Pokud nesouhlasí kontrolní součet přes paměť EEPROM, vrací Nak.

2.3. Adresa uzlu

kód příkazu	data	směr přenosu
2	W	R/W

Čtení adresy modulu (uzlu), nebo zápis nové adresy modulu odpovídající adrese Lecom protokolu. Pro akceptování nové adresy musí být nastavena konfigurační propojka (zkratován Jumper - svítí trvale LED). Nová adresa se automaticky uloží do konfigurace v EEPROM. Všechny moduly vždy akceptují adresu 0. Pokud se nastaví cílová adresa 0, zápis se provede, ale nepotvrzuje se.

2.4. Čtení vstupního portu IN0, IN1

kód příkazu	data	směr přenosu
12 (0Ch)		W R

Čtení stavu vstupního portu P0, P1. Na zápis vrací NAK.

2.5. Zápis na výstupní port OUT0, OUT1

kód příkazu	data	směr přenosu
13 (0Dh)		W R/W

Zápis nastaví výstupní port OUT0 a OUT1

Čtení přečte stav výstupních záchytných registrů.

2.6. Nastavení měřicího vstupu čítače 0

kód příkazu	data	směr přenosu
20 (14h)		W R/W

Přečte nebo nastaví, na kterém vstupu se bude čítačem 0 měřit. Číslo vstupu je v rozsahu 1..12. Zadávaný vstup musí být příkazem Směr portu nastaven na vstup, čítač bude změny na výstupu ignorovat. Příkaz současně ruší činnost předchozího nastavení čítače.

2.7. Předdělič pro čítač 0

kód příkazu	data	směr přenosu
21 (15h)		W R/W

Čtení nebo zápis hodnoty předděliče čítače číslo 0.

2.8. Start měření čítače 0

kód příkazu	data	směr přenosu
22 (16h)		W R/W

Čtení přečte status čítače 0. Jednička znamená, že čítač dočítal. Zápis způsobí start čítače. Při zápisu jsou data ignorována. V případě, že čítač běží, není příkaz start čítače akceptován (odpovídá NAK).

2.9. Čtení hodnoty čítače 0

kód příkazu	data	směr přenosu
23 (17h)		R Read

Přečte data z čítače 0. Zápis je ignorován, vrací NAK. Data jsou dekadická v rozsahu <0;8000000>.

2.10. Nastavení měřicího vstupu čítače 1

kód příkazu	data	směr přenosu
24 (18h)		W R/W

Přečte nebo nastaví, na kterém vstupu se bude čítačem měřit. Číslo vstupu je v rozsahu 1..12.

2.11. Předdělič pro čítač 1

kód příkazu	data	směr přenosu
25 (19h)		W R/W

Čtení nebo zápis hodnoty předděliče čítače číslo 1.

2.12. Start měření čítače 1

kód příkazu	data	směr přenosu
26 (1Ah)		W R/W

Čtení přečte status učítače 1. Jednička znamená, že čítač dočítal. Zápis způsobí start čítače, data jsou ignorována. Z případě, že čítač běží, není další start čítače akceptován (vrací NAK).

2.13. Čtení hodnoty čítače 1

kód příkazu	data	směr přenosu
27 (1Bh)		R Read

Přečte data z čítače 1. Zápis je ignorován, vrací NAK.

3. Konfigurace po startu

Konfigurace je uložena v paměti EEPROM na desce PIC03 v následujícím pořadí:

BYTE	NÁZEV	VÝZNAM
0	vEConf	status procesoru
1	vENode	adresa modulu
2	vESet1	výstupy 1..8 (směr)
3	vESet2	výstupy 9..12 (směr)
4	vEObsah1	výstupy 1..8 (obsah)
5	vEObsah2	výstupy 9..12 (obsah)

Význam bitů bytu vEConf je následující:

BIT	NÁZEV	VÝZNAM
0,1	sSPort	00..po resetu nastaví obsah portů podle EEPROM
		01.. po resetu nastaví obsah portů na 1
		10.. po resetu nastaví obsah portů na 0
		11..obsah portů zůstane nezměněn
7	sPDown	1..došlo k výpadku napájení

4. Komunikační protokol

Pro komunikaci nadřazeného s modulem PIC se používá podmnžina LECOM-Protokolu (DIN 66019, ISO 1745, ANSI X3,28). Tento protokol používá firma Lenze pro komunikaci s měniči motorů. Moduly PIC komunikují důsledně jako slave. Oproti protokolu Lecom se pro komunikaci s PIC moduly nepoužívají datové položky typu string, místo reálných čísel se používají čísla přirozená a nula v rozsahu <0;8 000 000>. Adresy modulů a code příkazů jsou pouze v rozsahu <0;99>.

Fyzická komunikace probíhá po sběrnici RS485 zapojené jako dvoudrát. Přenosové parametry jsou: 9600 Bd, 8 bitů, bez parity, 1 stop bit. U nově vyrobených modulů PIC je většinou nastaveno Node na 99. Všechny moduly PIC reagují na Node 0, ale nedávají odpověď.

4.1. Požadavek na zápis

Master posílá typ zápisového požadavku a data na zápis do zařízení.

Protokol: EOT, AD1, AD2, STX, C1, C2, V₁, ... V_n, ETX, BCC
EOT je ASCII znak 04h

AD1 je horní část adresy zařízení odpovídající desítkám. Obsahuje číslo 0..9 v ASCII reprezentaci.

AD2 je dolní část adresy zařízení odpovídající jednotkám. Obsahuje číslo 0..9 v ASCII reprezentaci.

STX je ASCII znak 02h.

C1,C2 jsou dva ASCII znaky určující typ parametru. Typ parametru se v našem případě jmenuje Code příkazu. Typ parametru odpovídá číslu v rozsahu <0;99>. C2 určuje dekadicky jednotky typu parametru, C1 určuje dekadicky desítky typu parametru. C1, C2 mohou obsahovat znaky '0' .. '9'.

V₁...V_n je parametr, který může být ve formátu:

VD₁..VD₇ = přirozené číslo a nula. Kde VD_n je ASCII reprezentace decimální číslice. Rozsah předávaného čísla je v intervalu <0;8 000 000>.

'H' **VH₁..VH₄** = číslo v hexadecimálním formátu. Kde VH_n jsou dva nebo čtyři znaky z ASCII znaků '0'..'9', 'A', 'B', 'C', 'D', 'E', 'F'. VH₁ je číslo nejvyššího šestnáctkového řádu VH₄ je nižšího řádu.

V našem případě jsou v parametrech V₁ .. V₃ uložena data příkazu.

ETX je ASCII znak 03h.

BCC je byte příčné parity = xor <C1 .. ETX>.

Zařízení jako odpověď na požadavek zápisu posílá masterovi znak **ACK** = 06h, který oznamuje úspěšnou operaci, nebo znak **NAK** = 15h, který oznamuje operaci neúspěšnou.

4.2. Požadavek čtení

Master posílá typ požadavku na čtení ze zařízení.

Protokol: EOT, AD1, AD2, C1, C2, ENQ

EOT je ASCII znak 04h.

AD1 je horní část adresy zařízení odpovídající desítkám. Obsahuje číslo 0..9 v ASCII reprezentaci.

AD2 je dolní část adresy zařízení odpovídající jednotkám. Obsahuje číslo 0..9 v ASCII reprezentaci.

C1,C2 jsou dva ASCII znaky určující typ parametru. V našem případě je zde uložen kód příkazu. Typ parametru odpovídá číslu v rozsahu <0;99>. C2 určuje dekadicky jednotky typu parametru. C1 určuje dekadicky desítky typu parametru. C1, C2 mohou obsahovat znaky '0' .. '9'.

ENQ je ASCII znak 05h.

Zařízení odpovídá na požadavek na čtení dat jednou z následujících tří typů zpráv:

1. Žádost jsem správně přijal a dekodoval a posílám požadovaná data:

protokol: **STX**, C1, C2, V₁ .. V_n, **ETX**, **BCC**

2. Při chybě se vrací pouze **NAK**.

STX je ASCII znak 02h.

C1,C2 jsou dva ASCII znaky určující typ parametru. Typ parametru odpovídá číslu v rozsahu <0;99>. C2 určuje dekadicky jednotky typu parametru a může obsahovat znaky '0' .. '9'. C1 určuje dekadicky desítky typu parametru a může obsahovat tytéž znaky jako C1.

V₁...V_n je parametr, který může být ve formátu:

VD₁..VD₇ = reálné číslo. Kde VD_n je asci reprezentace decimální číslice, nebo znak '-', nebo znak '.'. Rozsah předávaného čísla je v intervalu <-32767;+32768>. Číslo nesmí končit desetinou tečkou, za kterou není další číslice.

'H', VH₁..VH₄ = číslo v hexadecimálním formátu. Kde VH_n jsou dva nebo čtyři znaky z asci znaků '0'..'9', 'A', 'B', 'C', 'D', 'E', 'F'. VH₁ je číslo nejvyššího šestnáctkového řádu VH₄ je nenižšího řádu.

'S', S₁..S₄ = string. Kde S_n je asci tištiteľný znak <' ; '~>.

ETX je ASCII znak 03h.

BCC je byte příčné parity = xor <C1 .. ETX>.

EOT je ASCII znak 04h.