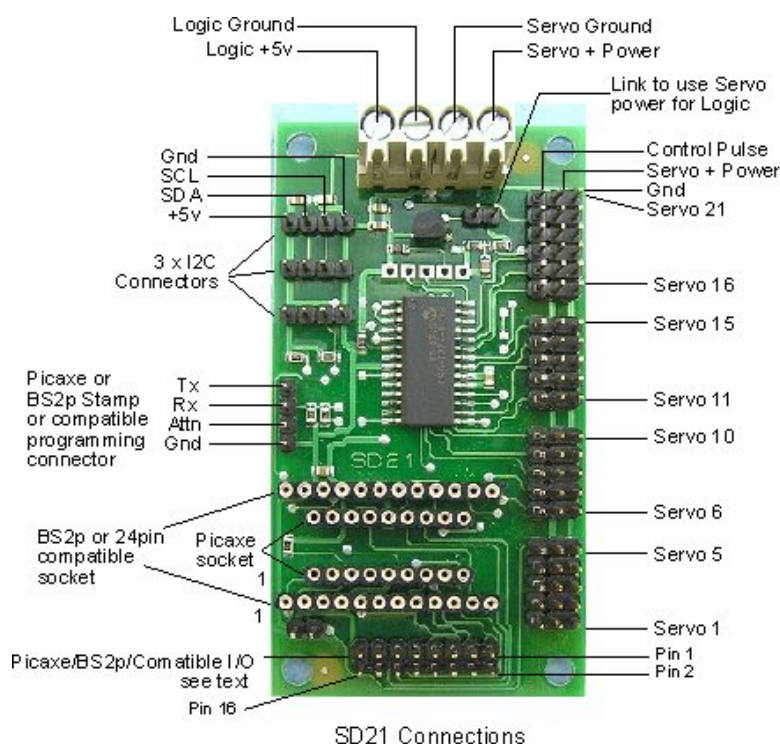


## SD21

### Ovládací modul pro 21 servomotorů s I2C interface

#### Technické údaje

SD21 je modul k ovládání modelářských servomotorů (serv). Umožňuje řídit polohu až 21 serv a přitom dodržuje periodu řídicích impulsů 20ms, bez ohledu na počet použitých kanálů a polohy serv (na rozdíl od řadiče SD20, který periodu podle potřeby prodlužuje). SD21 dokáže řídit nejenom polohu, ale i pohyb jednotlivých serv. Ovládá se příkazy posílanými po sběrnici I2C procesoru PIC18F2220. Na desce modulu jsou celkem tři konektory sběrnice I2C, kterýkoliv může být použit k připojení nadřazeného systému. Druhou možností je zasunout přímo do patice některý z následujících kontrolérů – Picaxe-18X, BS2p, Atom, BX-24, čímž vznikne kompaktní ovládací systém.

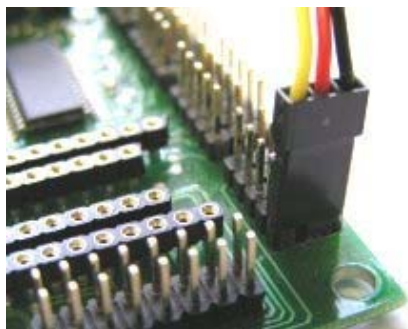


#### Napájení

Sd21 lze napájet dvěma způsoby. Buďto použít jeden zdroj 5V pro procesor a druhý zdroj 6V-7.2V pro napájení servomotorů. Tomuto způsobu dáváme přednost, čtyřpólová svorkovnice je k tomu připravena. Záporné póly obou zdrojů jsou na desce propojeny. Pokud nechceme použít dva zdroje (dvě baterie), lze modul napájet z jednoho zdroje (typicky o napětí 7.2V). Potom je třeba spojit kolíčky v blízkosti napájecího konektoru, což připojí napájení na stabilizátor 5V s nízkým úbytkem. Přívod napájení je potom připojen ke konektorům serv, v žádném případě k napájení procesoru. SD21 měří napájecí napětí baterie, které lze přečíst v jednom z registrů.

## Serva

Serva se zapojují přímo k modulu SD21, zemní přívod (na servech Hitec černý vodič) směrem k okraji desky.



## Basic Stamp BS2p nebo kompatibilní kontrolér

Na SD21 je připravena 24 kolíková patice pro BS2p nebo kompatibilní kontrolér, jako ATOM nebo BX-24. Vývod 1 BS2p směřuje k okraji desky, na opačnou stranu nežli jsou konektory pro serva. Porty P0(vývod 5) a P1(vývod 6) jsou použity jako řídicí signály I2C sběrnice SDA a SCL. Jedině tyto vývody jsou propojeny v modulu, ostatní jsou vyvedeny na 16 kolíkový konektor.

	16	15	
Gnd	□	□	Gnd
P15	□	□	P14
P13	□	□	P12
P11	□	□	P10
P9	□	□	P8
P7	□	□	P6
P5	□	□	P4
P3	□	□	P2
	2	1	

16 pin Header - Top View



## Procesor Picaxe

18 kolíková patice je připravena pro PICAXE-18X. Výstupy 1 a 4 jsou použity jako signály I2C sběrnice a zbylé vstupy a výstupy jsou opět vyvedeny na 16 kolíkový konektor. Vývod 1 Picaxe je orientován směrem k okraji desky, na opačnou stranu nežli jsou konektory pro serva.

	16	15	
Gnd	□	□	Gnd
N/C	□	□	N/C
OUT7	□	□	OUT6
OUT5	□	□	OUT3
OUT2	□	□	OUT0
INP7	□	□	INP6
INP2	□	□	N/C
INP1	□	□	INP0
	2	1	

16 pin Header - Top View



## Řadič serv

Výkonným obvodem SD21 je naprogramovaný procesor PIC18F2220. Ten je ovládán po sběrnici I2C na adrese 0xC2 (\$C2) jedním z výše uvedených způsobů – procesorem v patici modulu nebo připojeným ke konektoru I2C sběrnice. K řízení každého serva slouží tři vnitřní registry, ovládající rychlost a polohu (horní a dolní byte).

<i>Registr</i>	<i>Servo</i>	<i>Funkce</i>	<i>Registr</i>	<i>Servo</i>	<i>Funkce</i>	<i>Registr</i>	<i>Servo</i>	<i>Funkce</i>
0	1	Rychlost	24	9	Rychlost	48	17	Rychlost
1	1	Spodní byte	25	9	Spodní byte	49	17	Spodní byte
2	1	Horní byte	26	9	Horní byte	50	17	Horní byte
3	2	Rychlost	27	10	Rychlost	51	18	Rychlost
4	2	Spodní byte	28	10	Spodní byte	52	18	Spodní byte
5	2	Horní byte	29	10	Horní byte	53	18	Horní byte
6	3	Rychlost	30	11	Rychlost	54	19	Rychlost
7	3	Spodní byte	31	11	Spodní byte	55	19	Spodní byte
8	3	Horní byte	32	11	Horní byte	56	19	Horní byte
9	4	Rychlost	33	12	Rychlost	57	20	Rychlost
10	4	Spodní byte	34	12	Spodní byte	58	20	Spodní byte
11	4	Horní byte	35	12	Horní byte	59	20	Horní byte
12	5	Rychlost	36	13	Rychlost	60	21	Rychlost
13	5	Spodní byte	37	13	Spodní byte	61	21	Spodní byte
14	5	Horní byte	38	13	Horní byte	62	21	Horní byte
15	6	Rychlost	39	14	Rychlost	63	-	
16	6	Spodní byte	40	14	Spodní byte	64	-	Verze software
17	6	Horní byte	41	14	Horní byte	65	-	Napětí baterie
18	7	Rychlost	42	15	Rychlost			
19	7	Spodní byte	43	15	Spodní byte			
20	7	Horní byte	44	15	Horní byte			
21	8	Rychlost	45	16	Rychlost			
22	8	Spodní byte	46	16	Spodní byte			
23	8	Horní byte	47	16	Horní byte			

## Poloha serva

Poloha serva (v pořadí dolní pak horní byte) je vyjádřena 16 bitovým číslem, které přímo udává délku pulsu v mikrosekundách. Například hodnota 1500 (1500 $\mu$ s čili 1.5ms) nastaví většinu serv do střední polohy. Běžný rozsah pulsů je od 1000 $\mu$ s (1ms) do 2000 $\mu$ s (2ms), ale obvykle serva zpracují i o něco kratší nebo delší řídicí pulsy. Například servo Hitec HS311 umožňuje nastavovat polohu od 800 to 2200. Při využívání širšího rozsahu je však třeba opatrnosti, aby se servo nezastavilo o svoje vnitřní dorazy. Registry polohy mohou být po I2C sběrnici také zpětně čteny a pokud se pohyb serva děje řízenou rychlostí, udávají okamžitou polohu na cestě ke konečné zadané hodnotě (viz níže).

## Rychlost serva

Registr rychlosti udává, jakou rychlostí se servo pohybuje do zadané polohy. Pokud je nastaven rychlost nula šířka řídicích pulsů je okamžitě změněna na novou hodnotu. Rychlost pohybu je potom daná vlastnostmi serva, jeho zatížením a napájecím napětím. Všechny registry rychlosti jsou po zapnutí napájení nastaveny na nulu, takže serva se pohybují maximální možnou rychlostí. Pokud je registr rychlosti nastaven na jinou hodnotu nežli nula, potom tato hodnota udává o kolik mikrosekund se změní šířka každého následujícího pulsu (každých 20ms), dokud není dosažena konečná hodnota.

Například pokud zadáme pohyb z polohy 1000 do 2000 rychlostí 10, potom bude trvat 2 sekundy. Vzorec pro celkový čas pohybu je:

$$((\text{Konečná poloha} - \text{Počáteční poloha}) / \text{Rychlost}) * 20\text{ms}$$

Několik příkladů:

<i>Počáteční poloha</i>	<i>Konečná poloha</i>	<i>Rychlost</i>	<i>Doba pohybu</i>
2000	1000	10	2 000ms (2.0sec)
1000	2000	10	2 000ms (2.0sec)
1000	2000	1	20 000ms (20sec)
1000	2000	100	200ms (0.2sec)
1234	1987	69	220ms (0.22sec)

## Další registry

K ovládání serv postačí výše uvedené registry, avšak aby bylo ovládání snadnější pro kontroléry s omezenou velikostí paměti, jako je Picaxe, je tu ještě další sada registrů (63-83), které umožňují nastavit polohu zapsáním jediného byte namísto dvou. Tyto registry nejsou fyzicky implementovány, takže je nelze číst. Při zápisu je hodnota vynásobena 6 a přičten offset 732, výsledek je zapsán do 16 bitového registru polohy. Tímto způsobem lze zapsat polohu od 732 ( $0*6+732$ ) do 2262 ( $255*6+732$ ) s krokem 6 $\mu$ s. Tyto registry se nazývají Base registry.

Příslušný vzorec je :

$$\text{Doba pulsu} = \text{BaseReg} * 6 + 732\mu\text{s}$$

Kromě Base registrů existují i Positivní registry s adresami (84-104) a Negativní registry s adresami (105-125). Zapsáním do těchto registrů se hodnota přičte k Base registru nebo odečte od Base registru. Výsledek se potom opět vynásobí 6 a přičte offset 732, což poskytne šířku pulsu v mikrosekundách. Ani tyto registry nejsou fyzicky implementovány a nelze je tudíž číst.

**Vzorce:**

$$\text{Doba pulsu} = (\text{BaseReg} + \text{PosReg}) * 6 + 732$$

nebo

$$\text{Doba pulsu} = (\text{BaseReg} - \text{NegReg}) * 6 + 732$$

<i>Servo</i>	<i>Base Reg</i>	<i>Pos Offset Reg</i>	<i>Neg Offset Reg</i>
1	63	84	105
2	64	85	106
3	65	86	107
4	66	87	108
5	67	88	109
6	68	89	110
7	69	90	111
8	70	91	112
9	71	92	113
10	72	93	114
11	73	94	115
12	74	95	116
13	75	96	117
14	76	97	118
15	77	98	119
16	78	99	120
17	79	100	121
18	80	101	122
19	81	102	123
20	82	102	124
21	83	104	125

## Souhrn registrů

Pro nejpresnější řízení slouží 16 bitové (fyzicky implementované) registry, které určují šířku pulsu přímo v  $\mu\text{s}$ . Kontroléry s menší pamětí nebo při menších požadavcích na přesnost lze použít též 8 bitové registry. Pozitivní a negativní registry se uplatní například u chodících robotů, kde lze řídit serva na obě strany vzhledem ke střední poloze. Na adrese <http://www.robot-electronics.co.uk/html/eh2bs2p.htm> lze nalézt příklad ovládání šestinožného robota Lynxmotion EH2 kontrolérem Stamp BS2p pomocí 16 bitových registrů. Stejného cíle dosahuje Picaxe s použitím 8 bitových registrů.

## Verze software

Registr 64 obsahuje číslo verze interního software SD21. V době psaní dokumentace to bylo číslo 3.

## Napětí baterie

Registr 65 obsahuje napětí baterie v jednotkách 39mV, maximum je 10V. Baterie s napětím 7.2V dává hodnotu asi 184, při napětí baterie 6V přečteme kolem 154. Tato hodnota je změřena každých 20ms bez ohledu na přečtení registru.

## Adresa

Servo modul SD21 má pevně přiřazenu adresu na I2C sběrnici 0xC2.

## Příklad kódu

Použití Stamp BS2p k řízení serva, používá 16 bitové registry.  
Servo vykonává cyklické pohyby mezi dvěma polohami:

```
'{$STAMP BS2p}

SDA      CON 0      ' SDA on pin0, SCL on pin1
SD21     CON $C2    ' SD21 I2C address
Servo1   CON 0      ' register address of servo1 speed reg (followed by pos low/pos high)
Speed    CON 0      ' maximum speed
Servo1p  CON 1800   ' Right position
Servo1n  CON 1200   ' Left position

Servo VAR W0

Loop:
Servo = Servo1p
I2COUT SDA, SD21, Servo1, [Speed, Servo.LOWBYTE, Servo.HIGHBYTE]
PAUSE 300
Servo = Servo1n
I2COUT SDA, SD21, Servo1, [Speed, Servo.LOWBYTE, Servo.HIGHBYTE]
PAUSE 300
GOTO Loop
```

Tentýž příklad pro Picaxe používající 8 bitové registry:

```
Servo1   = 63          ' servo 1 base register
Servo1p  = 84          ' servo 1 positive offset register
Servo1n  = 105         ' servo 1 negative offset register
Base     = 128         ' centre position
Offset   = 50          ' +/- 50 from centre position

ProgStart:
i2cslave $c2, i2cslow, i2cbyte      ' setup i2c port for servo controller
writei2c Servo1, (Base)

Loop:
writei2c Servo1p, (Offset)
pause 300
writei2c Servo1n, (Offset)
pause 300
goto Loop
```