

# hserin

---

Platí pro PICAXE 08M2, 14M2, 18M2, 20M2, 28X1, 28X2, 40X1, 40X2

## Syntaxe:

**Pro PICAXE řady X2:**

**HSERIN spaddress, count {,(qualifier)}**

**HSERIN [timeout, address], spaddress, count {,(qualifier)}**

**Qualifier** – jedna proměnná nebo konstanta datového typu byte (0 až 255), která musí být přijata před ukládáním hodnot do paměti.

**Spaddress** – první adresa zápisníkové paměti, kam budou ukládána data

**Count** – počet bytů, které mají být načteny

**Timeout** – proměnná nebo konstanta definující čas čekání na signál v milisekundách

**Address** – adresa, na níž program odskočí v případě překročení časového limitu

**Pro PICAXE řady M2:**

**HSERIN var**

**Var** – proměnná typu byte, do níž bude přijat jeden byte

## Popis:

Sériový vstup dat přes pin spojený s interním hardwarovým převodníkem. Data jsou přenášena ve formátu (8 data bitů, bez parity, 1 stop bit). Není možné ho použít na ostatních pinech.

Nastavení přenosových vlastností se provádí před prvním použitím příkazu hserin příkazem hsersetup.

Přenosová rychlost je mnohem vyšší než u příkazu serin. Proto není možné použití pomocných funkcí, např. „#“, ale „surová“ data jsou ukládána rovnou do paměti. Po kompletním přenosu je musí obsloužit uživatelský program.

U PICAXE řady X2 se data ukládají na pozadí běhu programu přímo do paměti.

U PICAXE řady M2 je využito dvojbytového zásobníku FIFO, do nějž jsou na pozadí stále ukládána přijatá data. Při použití příkazu hserin je vyšší hodnota uložena do proměnné. Pokud nejsou přijata nová data, hodnota se nezmění, pokud jsou uloženy dvě hodnoty, je třeba příkaz hserin volat opakovaně. V případě přetečení FIFO zásobníku jsou starší data ztracena.

## Související příkazy:

- serin
- serrxd
- hsersetup
- hserout

### **Příklad pro PICAXE řady X2:**

odeslání a přijetí 4 bytů po sériové lince rychlostí 19 200 baud

```
hsersetup B19200_16, %00 ;nastavení přenosu 19 200 baud / 16 MHz
```

```
main:
```

```
hserin [1000,main],0,4 ; přijmi 4 byty do paměti sp
```

```
ptr = 0 ; vynuluj ukazatel pozice v sp
```

```
hserout 0,(@ptrinc,@ptrinc,@ptrinc,@ptr) ; odešli 4 byty z sp
```

```
goto main ; zpět na „main“
```