



evropský  
sociální  
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,  
MLÁDEŽE A TĚLOVÝCHOVY

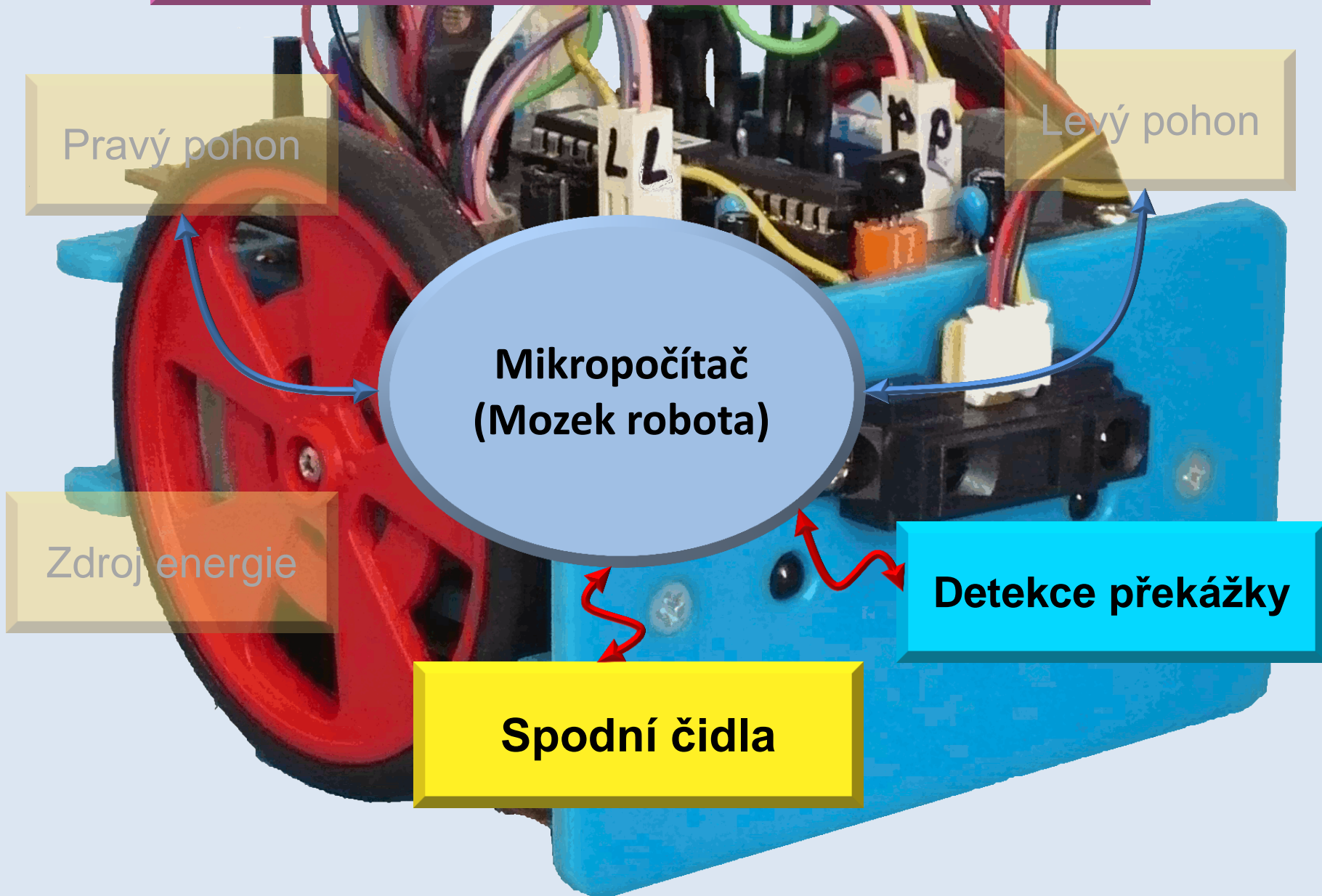


OP Vzdělávání  
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

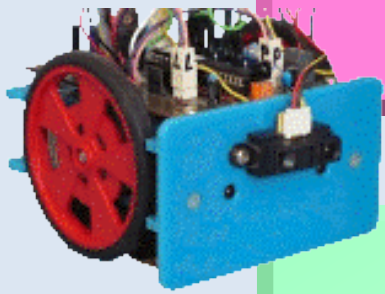
<b>Škola</b>	Střední průmyslová škola a Vyšší odborná škola, Hrabákova 271, Příbram
<b>Číslo projektu</b>	CZ.1.07/1.5.00/34.0556
<b>Číslo a název šablony klíčové aktivity</b>	III / 2 = Inovace a zkvalitnění výuky prostřednictvím ICT
<b>Tematická oblast</b>	Mikroprocesorová technika
<b>Název sady</b>	Obsluha základních periférií mikropočítačem
<b>Téma</b>	Robot hraje „MINISUMO“
<b>Anotace</b>	Robotika, autonomní chování robota v ohraničeném prostoru. Programová obsluha optického čidla, reakce na překážku, přerušení programu.
<b>Autor</b>	Ing. Josef Řehout
<b>Den vytvoření</b>	27.2.2014
<b>Den ověření</b>	11.3.2014
<b>Označení materiálu</b>	VY_32_INOVACE_RE_ELT_20

# Robot hraje minisumo



# Robot hraje minisumo





# Robot hraje minisumo

**Otevři sešit**

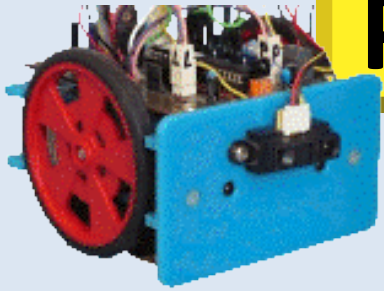
**Nadepiš**

**„Robot hraje minisumo“**

**Dělej si poznámky z následujících  
stran prezentace**

**Jen tak budeš schopen robota  
správně naprogramovat**

# Princip detekce okraje arény

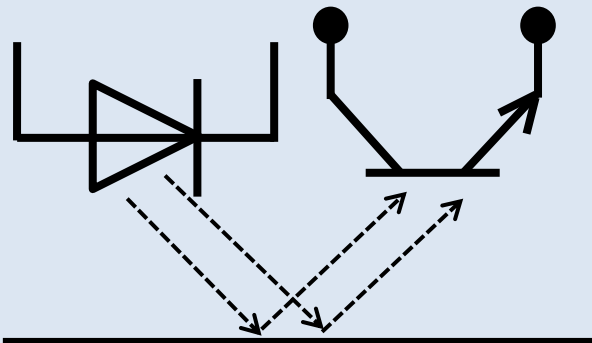


## Spodní optočidlo

- LED vysílá světelný paprsek
- Fototranzistor detekuje odraz od podložky
- Světlá podložka – silný odraz – logická 0
- Tmavá podložka – slabý odraz – logická 1
- Čidlo pracuje s infrazářením – minimalizace falešných detekcí jiných zdrojů světla

Výstup čidla – kolektor fototranzistoru

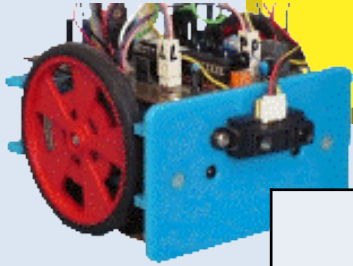
PINx.n



## Princip čidla

- IRLED je trvale napájena
- Kolektor je napájen přes rezistor
- Emitor je uzemněn
- Bez odrazu je tranzistor zavřen – logická 1
- Při odrazu je tranzistor otevřen – logická 0

# Připojení čidel okrajů arény k mikropočítači PICAXE 20M2



## Čidla podložky (spodní čidla)

<b>LKČ</b>	<b>LSČ</b>	<b>PSČ</b>	<b>PKČ</b>
<b>PINB.3</b>	<b>PINB.1</b>	<b>PINB.0</b>	<b>PINB.2</b>

Pozn.: k detekci okraje arény použijeme pouze krajní čidla

### Vysvětlivky zkratk v tabulce

**LKČ** – levé krajní čidlo

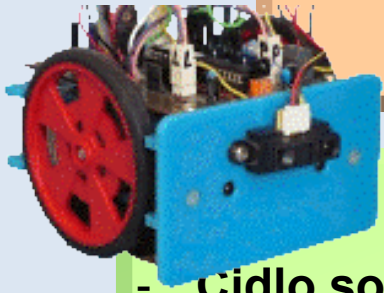
**LSČ** – levé střední čidlo

**PSČ** – pravé střední čidlo

**PKČ** – pravé krajní čidlo

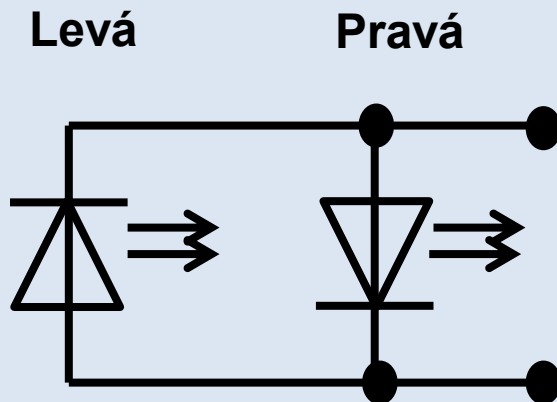
**Stav čidel je vhodné načíst do jednobitových proměnných**

# Detekce soupeře



## Princip vytvoření signálu čidla

- Čidlo soupeře (překážky) je tvořeno třemi prvky. Levou a pravou IRLED a detektorem IR signálu 36 kHz.
- Výstup C.5 = frekvence 36 kHz je nepřetržitě generována příkazem PWMOUT, a lze nastavit dosah čidla.
- Výstup C.4 = 5 V aktivuje levou IR LED
- Výstup C.4 = 0 V aktivuje pravou IR LED
- Použité výstupy je nutno respektovat podle skutečného zapojení robota.

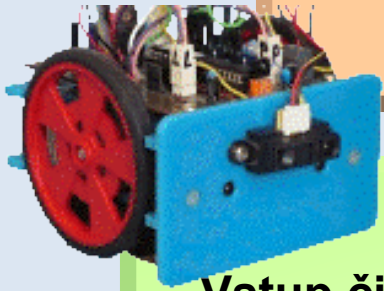


PINC.5, PWM 36 kHz



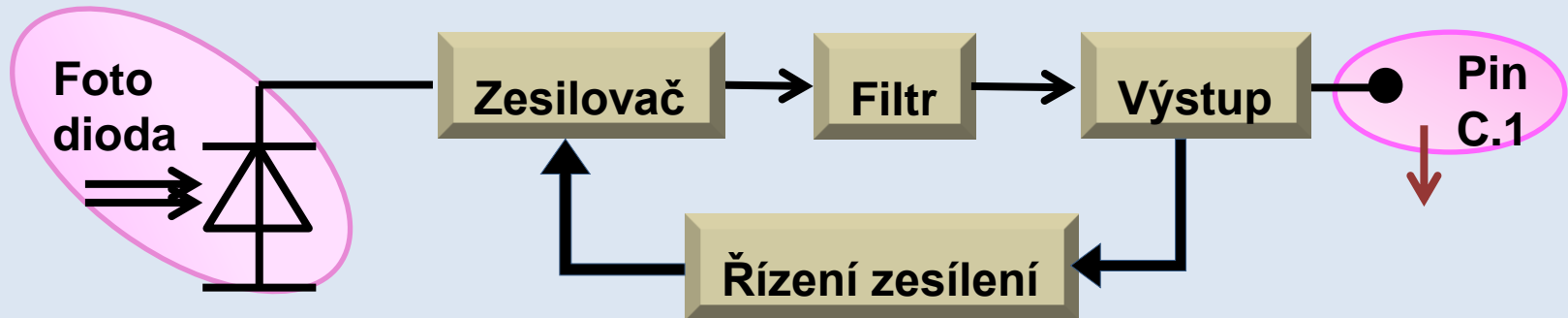
PINC.4: HIGH ... Levá LED  
LOW ... Pravá LED

# Detekce soupeře



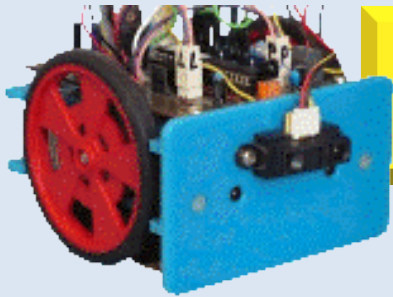
## Čtení stavu čidla soupeře (překážky)

- Vstup čidla na pinC.1 (nebo jiný podle konfigurace robota)
- Při stavu „1“ soupeř není v dosahu čidla
- Při stavu „0“ je soupeř v dosahu čidla
- Nutno respektovat skutečný port mikropočítače, kam jsou všechny části čidla připojeny nebo jiný typ čidla.



Podrobněji o čidlech podložky a překážky  
viz předchozí lekce

# Pravidla hry „MINISUMO“, aréna



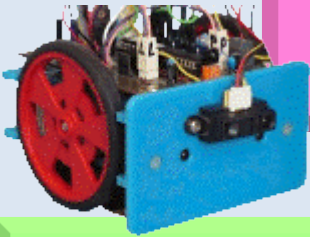
Bílý okraj (D = 80 cm)  
Černá plocha



Čidla jsou jen vpředu, robot se nemůže k okraji arény přiblížit pohybem vzad.

## Pravidla „MINISUMO“

- start z náhodné pozice
- pohyb s cílem najít soupeře a vytlačit jej mimo arénu
- nesmí z arény sjet ani při vytlačení soupeře
- při detekci soupeře musí viditelně zrychlit
- hra končí vytlačení jednoho ze soupeřů
- po devadesáti sekundách od startu se musí robot automaticky zastavit



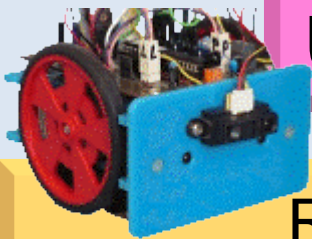
## Program pro hru „MINISUMO“

**Vytvořte program pro robota ve hře „minisumo“ podle výše uvedených pravidel a následujícího upřesnění:**

- Robot se rozjede 2 s po stisknutí tlačítka
- Použití vnitřního času k zastavení robota po 90 s od startu
- Použití přerušení pro detekci soupeře
- Přerušeni vysvětlíme dále

**Time = vnitřní hodiny mikropočítače**

- Počítá se od okamžiku zapnutí napájení
- Do proměnné typu word (16 bitů) se načítá čas po jedné sekundě
- Tato proměnná má název time (pro PICAXE 20M2)
- Disabletime ;tímto příkazem hodiny vynulujeme a zastavíme
- Enabletime ;tímto příkazem hodiny rozběhneme od 0
- Proměnnou time můžeme načíst do jiné proměnné



# Ukázka programu interrupt - přerušeni

Robot jezdí vpřed a vzad před překážkou

```
;nastavení 36kHz IRLED
pwmout c.5,27,20
low c.4      ;pravá IRLED
;nastavení přerušeni
setint %00000000, %00000010
;program
do
main:
  forward a
  forward b
loop
;obsluha přerušeni
interrupt:
  halt a
  halt b
  backward a
  backward b
  pause 235
  halt a
  halt b
  setint %00000000,%00000010
return
```

Přerušeni je  
nutno obnovit,  
jinak nově  
nenastane

setint a, b

- a ... stav vstupů, při kterém má nastat přerušeni
- b ... maska, vybírá platné vstupy
- V tomto případě je maskou nastaven druhý bit zprava a stav tohoto vstupu je 0.
- K přerušeni dojde při stavu nula na portu C.1, protože PICAXE 20M2 může nastavit přerušeni právě pouze na portu C.1 až C.5

Přerušeni nebere ohled na nic, ihned se vykonají příkazy podprogramu interrupt, který musí být ukončen příkazem return a umístěn na konci celého programu.

# Dosud procvičené příkazy (I)

start:	;návěstí (název části programu)
BUTTON pin, stav1, 255, mezera, proměnná, tlačítko, návěstí	;snímání stisku tlačítka s potlačením zákmitů
data adresa, (data, data)	;zápis dat do eeprom před začátkem programu
read adresa, registr1	;čtení dat z eeprom
write adresa, b1, w1	;zápis dat během programu
dirsb=%11111110	;aktivace portů B.7 až B.1 jako výstupních
pinsb=%01111110	;nastav výstupy portu b
forward A (B)	;motor A (B) vpřed
backward A (B)	;motor A (B) vzad
halt A (B)	;motor stop
for b6 = 1 to 5	;začátek cyklu
next b6	;konec cyklu podle proměnné b6
goto start	;skok
gosub podprogram	;jdi na podprogram
return	;návrat z podprogramu
high B.4	;výstup do 1
low B.4	;výstup do 0
toggle c.7	;přepni stav výstupu
if pinB.0 = 1 then suma	;podmíněný skok

# Dosud procvičené příkazy (II)

inc b1	;zvýšení obsahu proměnné b1 o 1
interrupt:	;návěstí podprogramu přerušení
setint vstupy, maska	;nastavení přerušení
pause 1000	;pauza v ms
poke 80,b1	; vlož proměnnou b1 do registru na adrese 80
peek 80,b1	; dej obsah registru 80 do proměnné b1
pwmout C.5, 27,6	; výstup, frekvence, šířka impulsu ; nastavení: PICAXE – Wizards - pwmout...
pulsout C.1, 10000	;výstupní pin, délka impulsu v milisekundách
random w5	;generování náhodného čísla (0 až 65535)
readadc B.2, b8	;AD převod z pinu b.2 ulož do proměnné b8
readadc10 B.2,w8	;AD desetibitový převod
select case b1	;Výběr možnosti podle hodnoty registru b1,
case < 77	;Když je b1 menší 77, splní se následující příkazy
endselect	;konec sekvence select case
serout pinout, baudmode, (data, data, data...)	;sériový výstup dat
servo B.4,75	;nastavení portu pro servomotor
servopos B.4,75	;natočení serva
Sleep y	;2,3s*y (konstanta 1 až 65535), nečinnost a nižší spotřeba
symbol K1=234	;konstanta (nezabírá místo proměnné)
symbol napeti=w12	;symbolický název proměnné w12

## Zdroje

- I. Revolution Education Ltd. PICAXE Manual, Section 1 - Getting Started, [www.picaxe.com/docs/picaxe\\_manual1.pdf](http://www.picaxe.com/docs/picaxe_manual1.pdf) [online] 5.10.2012, [cit. 27.2.2014]
- II. Revolution Education Ltd. PICAXE Manual, Section 2 – BASIC Commands, [www.picaxe.com/docs/picaxe\\_manual2.pdf](http://www.picaxe.com/docs/picaxe_manual2.pdf) [online] 5.10.2012, [cit. 27.2.2014]
- III. Revolution Education Ltd. PICAXE Manual, Section 3 - Microcontroller interfacing circuits, [www.picaxe.com/docs/picaxe\\_manual3.pdf](http://www.picaxe.com/docs/picaxe_manual3.pdf) [online] 5.10.2012, [cit. 27.2.2014]

Dostupné z <http://www.picaxe.com/Getting-Started/PICAXE-Manuals/>

Obrázky, programy – archiv autora



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ