

Přípravy na kroužek Mechatronika – KA6(A2d)

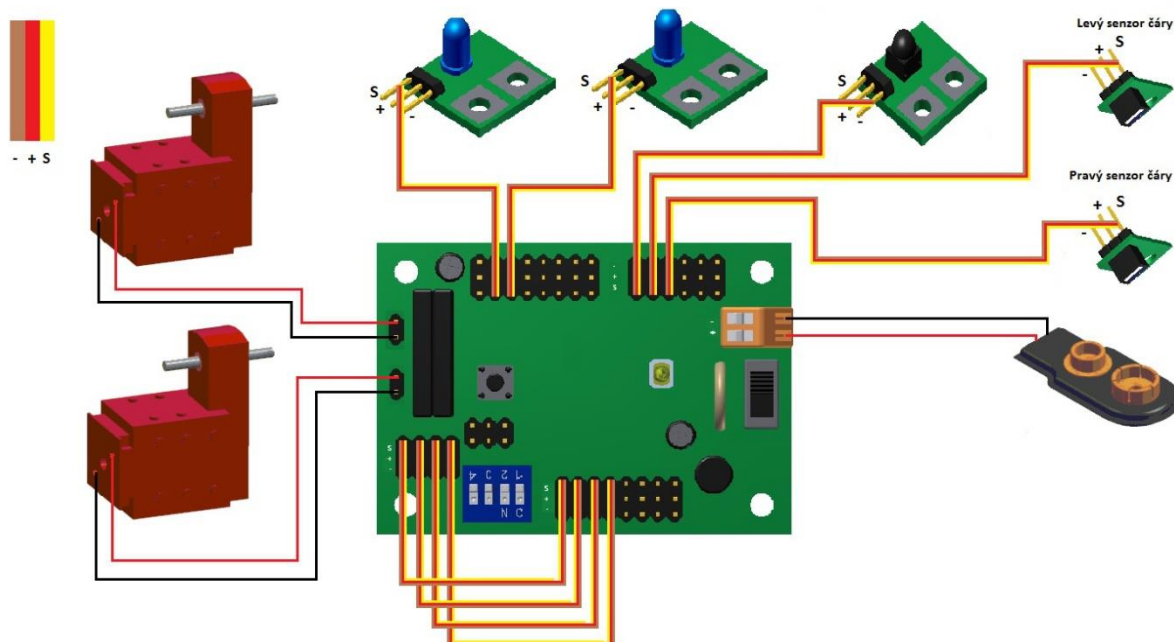
8. lekce – zapojení senzorů, dokončení vozítka, otestování funkčnosti

Zapojení senzorů

Pro dokončení stavby vozítka je nutné zapojit dvojici reflexních senzorů a IR čidla. Reflexní senzory slouží vozítku k detekci změny barvy povrchu, po němž se pohybuje, IR čidlo pak k detekci ostatních vozítek, která jsou v jeho dosahu.

Pro účely identifikace ostatním je vozítko vybaveno dvěma IR LED, které je nutné též zapojit.

Zapojení provedeme podle schématu ze stavebnice Merkur:



IR senzor zapojíme na PORTC0, dbáme při tom na správnou polaritu dle schématu (hnědý vodič je **záporný** pól napájení a oranžový pro signál mezi MCU a čidlem).

Levý reflexní senzor zapojíme na PORTC1, **pravý reflexní senzor** pak na PORTC2. Opět zkontrolujeme správnou polaritu a zapojení vodičů.

IR LED můžeme zapojit na PORTB1 a PORTB2 MCU.

Jako poslední provedeme nebo zkontrolujeme (pokud jsme zapojili minule) propojku mezi PORTD4-7 a regulátory motorů (viz obrázek). Motory necháme zapojené tak, aby polarita MOT2 byla obráceně oproti MOT1.

Tímto je stavba vozítka Merkur kompletně dokončena.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Otestování funkce vozítka

Nahrajeme na základní desku originální program Merkur MINISumo a otestujeme:

- chod motorů - po zapnutí vypínače se musí oba motory otáčet
- reflexní senzory - zakrytím jednotlivých senzorů prstem, v tomto okamžiku musí dojít ke změně chodu motoru. Otestujeme pravý i levý senzor.
- IR LED - nejlépe pomocí digitálního fotoaparátu (např. v mobilu), jelikož ten je schopen zaznamenat IR světlo.

Otestování IR senzoru necháme na později.

Pohyb vozítka vpřed

Založíme nový projekt v Atmel Studiu a v něm vytvoříme následující zdrojový kód:

```
#define F_CPU 8000000UL
#include <avr/io.h>

int main(void)
{
    DDRD = ((1<<PD4)|(1<<PD5)|(1<<PD6)|(1<<PD7));

    while(1)
    {
        PORTD = ((1<<PORTD5)|(1<<PORTD6));
    }
}
```

Program sestavíme, nahrajeme programátorem na základní desku a vyzkoušíme.

Princip funkce (popis programu):

Pomocí DDRD registru nastavíme příslušné kontakty MCU na výstup.

Na PORTD5 a PORTD6 nastavíme hodnotu 1, čím uvedeme vozítko do pohybu směrem vpřed (viz tabulky v lekci č. 7).

Podobným způsobem rozjedeme vozítko směrem vzad, pouze prohodíme hodnoty 1 na příslušných registrech PORTD.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

```
#define F_CPU 8000000UL
#include <avr/io.h>

int main(void)
{
    DDRD = ((1<<PD4)|(1<<PD5)|(1<<PD6)|(1<<PD7));

    while(1)
    {
        PORTD = ((1<<PORTD4)|(1<<PORTD7));
    }
}
```

Vypracoval Radek Zvěřina. Použité materiály: Merkur.

název projektu: **Cestou přírodovědných a technických oborů napříč Středočeským krajem**
registrační číslo: CZ.1.07/1.1.00/44.0011