

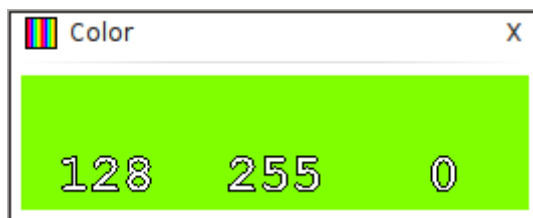
LORRIS TOOLBOX

sada nástrojů pro vývoj a řízení robotů

Lorris Toolbox je sada několika nástrojů, které mají společný cíl – pomáhat při vývoji, ladění a řízení robotů a jiných elektronických zařízení.

1. Analyzátor

- Soustřeďuje se na zobrazování dat z robota v grafické podobě
- Analyzátor pro zobrazování používá tzv. widgety – malá „okna“, která zobrazují určitou část dat
- Widgety mají individuální nastavení a uživatel si je může umístit na libovolné místo na pracovní ploše
- Lorris obsahuje několik typů widgetů, například *Číslo*, *Barva*, *Sloupcový bar*, *Kolo* (zobrazení úhlu v kružnici) či *Graf*.
- Pomocí widgetů lze sestavit rozhraní vyhovující prakticky jakémukoliv robotovi
- Analyzátor je ideální pro snadné zobrazování dat z prvků, u kterých není vhodné jako výstup použít čísla – například barevný senzor
- Některé widgety mohou posílat data i směrem do robota. Díky tomu je možné kromě zobrazování dat robota i ovládat
- Pozornost si zaslouží widget „script“. Uživatel v něm může napsat vlastní script, který zpracovává příchozí data. Script může využít ostatní widgety a další části Lorris, díky tomu lze zobrazit takřka jakákoliv data.



Příklad: widget „barva“

2. Uživatelské prostředí pro programátor Shupito

- Shupito je programátor mikrokontrolérů. Na jeden konec programátoru se připojí čip, na druhý počítač – bez programátoru nelze do mikrokontrolérů nahrát program.
- Lorris obsahuje uživatelské rozhraní pro ovládání Shupita – zapisování programu, čtení a mazání paměti čipu a programování pojistek

3. Terminál

- Klasický terminál - zobrazuje příchozí data jako text nebo vypisuje byty jako hexadecimální čísla.

4. Proxy mezi sériovým portem a TCP socketem

- Vytvoří server připojený na sériový port - k tomuto portu je pak možné se připojit odkudkoliv z internetu

PŘÍKLAD POUŽITÍ

Stavba robota, část 1: kostra

Prvním krokem při stavbě robota je obvykle postavení kostry s motory. Tyto widgety slouží pro ovládání robota pomocí joysticku připojeného k PC, tak lze otestovat funkčnost a chování motorů.

Widget Script, který přečte data z joysticku, vypočítá podle nich rychlost motorů a odešle ji do robota

```
Script
0: GreenAsia Inc.      USB Joystick
L: 187 R: 187
```

Nastavení joysticku:

- výběr joysticku
- maximální rychlost
- poloměr zatáčení
- prohození směru zatáčení

<div>Joystick setting [S]</div> <div>0: GreenAsia Inc. ▾</div> <div>Refresh joystick list</div> <div>Max speed: 255 ▾</div> <div>Turn speed 90 ▾</div> <div><input type="checkbox"/> Invert left/right</div>	<div>Speed left [S]</div> <div>187</div> <div>Speed right [S]</div> <div>187</div>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------

Výsledná rychlost

PŘÍKLAD POUŽITÍ

Stavba robota, část 2: senzory

Dalším stupněm vývoje je přidání všech senzorů, podle kterých se robot může orientovat. Tato část je konceptována jako pohled shora na robota - widget *Script* tvoří tělo robota, na každém rohu je tlačítko signalizující náraz do mantinelu, u obou kol je enkodér, vpředu jsou dva ultrazvukové měřáky vzdálenosti a barevný senzor.



PŘÍKLAD POUŽITÍ

Stavba robota, část 3: programování

Poslední je vývoj samotného programu robota. V tomto příkladu používám jednoduché „akce“, které robot postupně provádí.

Každá akce má 3 hlavní parametry - směr jízdy, kdy se má robot zastavit a co má vykonat když se zastaví na cílovém místě. Všechny akce je možno rovnou měnit, bez nutnosti přeprogramovat robota.

The screenshot shows a robot programming interface with several components and callouts:

- Widget Script, který obstarává veškerou funkčnost této části**: Points to the 'Program' window on the left, which displays a list of events and encoder status.
- Přidávání nové akce - dozadu, dopředu a na určitou pozici**: Points to the 'ev_head' window, specifically the 'Add back', 'Add front', and 'Insert to pos' buttons.
- Tlačítko, které spustí program**: Points to the 'Play events' button in the 'ev_head' window.
- Výběr strany hřiště**: Points to the 'Red side' dropdown menu in the 'ev_head' window.
- Směr pohybu robota**: Points to the 'Moveflags' dropdown menu in the first action row.
- Akce, kterou robot vykoná poté co dojezdí na určené místo**: Points to the 'SpecAct' dropdown menu in the first action row.
- Kdy se robot zastaví. Možnosti:**
 - když ujede určitou vzdálenost
 - když je stisknuto určité tlačítko
 - když robot dojezdí na určitou barvuPoints to the 'Stop act' dropdown menu in the first action row.

The 'Program' window shows the following text:

```
Event done 3
Encoder event 0 done
Event done 0
Encoder event 3 done
Event done 3
Encoder event 0 done
Event done 0
Encoder event 5 done
Event done 5
Encoder event 5 done
Event done 5
```

The 'ev_head' window shows the following actions:

1:	X	Moveflags:	MOVE NONE	SpecAct:	OPEN DOORS	Stop act:	None		
2:	X	Moveflags:	MOVE FORWARD	SpecAct:	CLOSE DOORS	Stop act:	Button	Button	DISC
3:	X	Moveflags:	MOVE NONE	SpecAct:	CLIMB	Stop act:	None		