

# BEÁGYAZOTT RENDSZEREK (C)

7. forduló



A kategória támogatója: Robert Bosch Kft.

## Ismertető a feladathoz

Most, hogy a Raspberry és az STM már tudnak kommunikálni, Gipsz Szabolcs a Raspberryn futó vezérlés kidolgozásához kezd. A vezérléshez a ROS (Robot Operating System) keretrendszert fogja használni. A Raspberryhez van kész és jól dokumentált ROS támogatás, így úgy sejtí, hogy kevés kódolás mellett inkább integrációs feladatokat kell majd megoldania. Segíts neki ezekben!

A megoldásban a következő adatlapok lesznek a segítségedre:

Raspberry Pi 4 adatlap: <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf>

ROS dokumentáció: <http://wiki.ros.org/>

RPLIDAR A2M8 adatlap:

[http://bucket.download.slamtec.com/d00df2d983fa1f47a24c56d489f4b4917773fb45/LD208\\_SLAMTEC\\_rplidar\\_datasheet\\_A2M8\\_v2.5\\_en.pdf](http://bucket.download.slamtec.com/d00df2d983fa1f47a24c56d489f4b4917773fb45/LD208_SLAMTEC_rplidar_datasheet_A2M8_v2.5_en.pdf)

RPLIDAR ROS csomag leírása: <http://wiki.ros.org/rplidar>

Felhasznált idő: 15:32/20:00

Elért pontszám: 1/27

## 1. feladat 0/6 pont

Szabolcs még csak most ismerkedik a ROS-sal. **Segíts neki az igaz állítások kiválasztásában!**

### Válaszok

- ☒ A ROS egy keretrendszer, ami hardver absztrakciót, processzek közötti kommunikációt, csomagkezelést és egyéb gyakran használt funkciókat, eszközöket nyújt.  
**Ez a válasz helyes, és meg is jelölted.**
- ☒ A ROS nodeok közötti kommunikáció aszinkron módon valósul meg.  
**Ez a válasz helytelen, de megjelölted.**
- ☒ A ROS topic-ok szigorúan típusosak.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☒ A csomagok függőségei a csomag manifeszt fájlokban vannak felsorolva.  
**Ez a válasz helyes, de nem jelölted meg.**
- ☒ Egy ROS node lehet egy processz, adathalmaz, konfigurációs fájl vagy bármi egyéb strukturáltan összegyűjtött hasznos információ összessége.  
**Ez a válasz helytelen, de megjelölted.**
- ☒ ROS nodeokat a rosrn és a roslaunch paranccsal is el lehet indítani.

Ez a válasz helyes, és meg is jelölted.

☐ A ROS keretrendszer indításához mindenképp a roscore parancsot kell először kiadni.

☒ Egy ROS megoldásban vegyesen is használhatunk roscpp és rospy könyvtárak segítségével C++-ban és Pythonban megírt ROS nodeokat.

Ez a válasz helyes, de nem jelölted meg.

☒ A ROS keretrendszerrel lehetőségünk van valós idejű rendszert építeni.

Ez a válasz helytelen, de megjelölted.

☒ ROS csomagot a catkin\_make paranccsal a catkin workspace könyvtárban lehet építeni.

Ez a válasz helyes, de nem jelölted meg.

☐ Az RViz és a Gazebo ugyanazt a célt szolgálja, de az utóbbi az újabb.

## Magyarázat

A ROS nodeok közötti kommunikáció lehet szinkron (ROS Service) és aszinkron (ROS Topics).

Amiről az *"Egy ROS node lehet egy processz, adathalmaz, konfigurációs fájlok vagy bármilyen egyéb strukturáltan összegyűjtött hasznos információ összessége."* pont szól, az a ROS csomag. A ROS node minden esetben egy processz.

A roscore parancsot nem szükséges kiadni: ha a roslaunch parancsot használjuk, ellenőrzi, hogy fut-e a ROS core, és ha nem, akkor automatikusan elindítja.

Bár a ROS lehetővé teszi gyors vezérlést igénylő komplex rendszerek építését is, az architektúrája miatt nem nevezhetjük valós idejű rendszernek.

Az RViz csupán 3D megjelenítésre képes, a Gazebo viszont egy teljes fizikai szimulációs környezetet ad.

## 2. feladat 1/1 pont

Szabolcs LIDAR használatával szeretné megvalósítani a robotfűnyíró navigációját. Ki is nézte a viszonylag olcsó, de várhatóan kellő teljesítményt nyújtó [RPLIDAR A2M8](#)-t, amihez van kész [ROS támogatás](#).

Milyen interfészen tudja csatlakoztatni a LIDAR-t a Raspberryhez? Válaszd ki a helyes választ!

### Válasz

☐ SPI

☐ I2C

☒ UART

Ez a válasz helyes, és meg is jelölted.

## Magyarázat

A LIDAR adatlapjáról leolvasható, hogy 3.3 V jelszintű UART-on keresztül kommunikál.

## 3. feladat 0/20 pont

Szabolcs szeretné az RViz névre hallgató grafikus megjelenítőben valós időben megjeleníteni a LIDAR által érzékelt környezetet.

Milyen parancsokat és milyen sorrendben kell futtatnia, hogy elérje célját?

Írd be a szövegdobozba a szükséges parancsok betűjeleit a megfelelő sorrendben szóköz, vessző, a záró zárójel és egyéb elválasztó karakterek nélkül! (tehát pl.: abcdefgh) Vigyázz, nincs mindegyik parancsra szükség!

a) cd ..

- b) `roslaunch rplidar_ros rplidarNode`
- c) `source /opt/ros/noetic/setup.bash`
- d) `git clone https://github.com/Slamtec/rplidar_ros.git`
- e) `sudo apt install ros-noetic-desktop`
- f) `mkdir -p ~/catkin_ws/src`
- g) `rviz -i rplidar_ros`
- h) `cd ~/catkin_ws/src`
- i) `catkin_make`
- j) `source devel/setup.bash`
- k) `roslaunch rplidar_ros view_rplidar.launch`
- l) `rosinit`

## Válaszok

### A helyes válasz:

ecfhdaijk

fechdaijk

efchdaijk

fhecdaijk

fehcdaijk

efhcdaijk

fhdecaijk

fhedcaijk

fehdcaijk

efhdcaijk

## Magyarázat

A szükséges parancsok megfelelő sorrendben:

`sudo apt install ros-noetic-desktop`

`source /opt/ros/noetic/setup.bash`

`mkdir -p ~/catkin_ws/src`

`cd ~/catkin_ws/src`

`git clone https://github.com/Slamtec/rplidar_ros.git`

`cd ..`

`catkin_make`

`source devel/setup.bash`

`roslaunch rplidar_ros view_rplidar.launch`

A ROS telepítése után inicializáljuk a ROS környezetet, aminek hatására elérhetővé válnak a ROS parancsok, majd egy catkin workspace-t hozunk létre, melybe letöltjük a LIDAR-hoz tartozó ROS csomagot. Ezt a catkin workspace könyvtárban futtatott catkin\_make paranccsal buildeljük, a devel/setup.bash sourceolásával elérhetővé tesszük a ROS számára, majd a roslaunch paranccsal elindítjuk.

A roslaunch parancs elindítja az rplidarNode-ot és az RViz-t is, így a b) és g) parancsokra nincs szükség, az utóbbi nem is egy jó parancs. Az l) pont béli rosinit nem létezik a ROS-ban.



[Legfontosabb tudnivalók](#) [Kapcsolat](#) [Versenyszabályzat](#) [Adatvédelem](#)

© 2023 Human Priority Kft.

KÉSZÍTETTE **cone**

Megjelenés

 Világos 