Диалогов интерфейс за управление на роботизирана ръка

Калоян Миронов

Dialog Interface for Controlling a Robotic Arm: *The paper describes the development of dialog interface for controlling a robotic arm. The project is demonstration of Java use in Robotics.*

Key words: *Java, Robotics, Interface, Robotic Arm.*

ВЪВЕДЕНИЕ

От създаването си през началото на 90-те, езикът Java става един от най-популярните езици за обектно-ориентирано програмиране. Успоредно с това се развиваха автоматизираните системи и в частност роботиката. С времето Java се доказа, като език, подходящ за разработване на роботизирани платформи.

Проектът, представен в доклада, е демонстрационен и има за цел да представи използването на Java за целите на роботиката.

Диалоговият интерфейс е част от проект за управление на роботизирана ръка (Фигура 1). Интерфейсът е изцяло изграден с помощта на езика Java в средата Eclipse.

Фигура 1

С цел по-добро представяне на проекта и поради ограничения, наложени от хардуера, ще разгледаме компонентите, които указват влияние върху програмния модел.

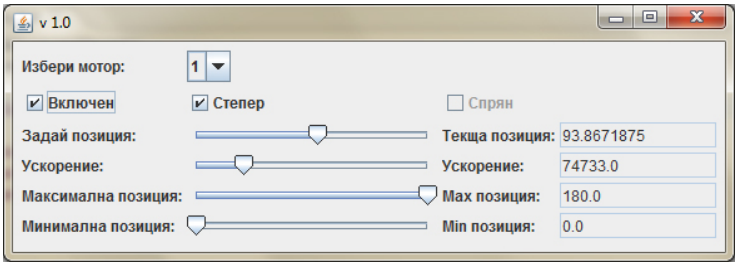
Хардуерната реализация на проекта е съставена от три основни компонента: шаси, контролер и серво-мотори. В Таблица 1 са разгледани гореспоменатите компоненти и са представени ограниченията, които те налагат на софтуерната реализация.

Таблица 1

Вид устройство	Вид ограничение
Серво контролер	Максимален брой поддържани мотори – 8
Серво-мотор	Обхват на действие – 180° (от -90° до 90°)

1. Характеристики на диалоговия интерфейс

Интерфейсът, управляващ една роботизирана платформа трябва да е лесен за използване, но в същото време и да предлага всички необходими за това функции. На Фигура 2 е показан диалоговият прозорец, чрез който се управляват хардуерните компоненти на робота.



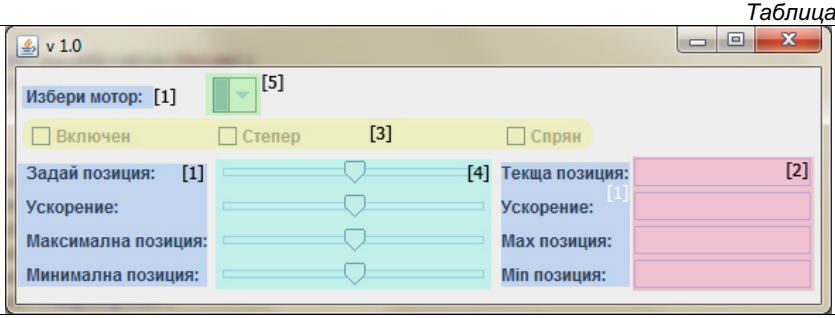
Фигура 2

1.1 Структура на диалоговия интерфвйс

Диалоговият прозорец е изграден с помощта на Swing компонентите: JLabel, JComboBox, JTextField, JSlider и JCheckBox. В Таблица 2 са представени Swing компонентите, тяхното разположение в диалоговия прозорец и функциите, които изпълняват. За улеснение, компонентите от всяка група са маркирани с различен цвят.

Таблица 2

Разположение



Swing компонент	Номер на група	Предназначение
JLabel	[1]	Съдържа текст, улесняващ използването на интерфейса.
JTextField	[2]	Визуализира актуалната стойност, използвана от функциите за управление.
JCheckBox	[3]	Променя режима на работа.
JSlider	[4]	Задава входни данни на управляващите функции.
JComboBox	[5]	Предлага избор на мотор.

1.2 Оформление

Разположението на компонентите в диалоговия прозорец е реализирано чрез MiG Layout [1]. Използването на този тип подредба се доближава до Flow Layout, но е значително по-гъвкаво и спомага за още по-добра обозримост на кода на програмата. В Таблица 3 е представена част от кода, използван в оформлението на диалоговия прозорец.

В Таблица 3 са представени първите два реда от оформлението на диалоговия интерфейс. При използването на MiG Layout, компонентите се разполагат поредово. За преминаване на нов ред използваме “wrap”.

Таблица 3

```
Container Panel = getContentPane();
Panel.setLayout(new MigLayout());
Panel.add(jLabel6);
Panel.add(servoCmb, "wrap");
Panel.add(engagedChk);
Panel.add(rampingChk);
Panel.add(stoppedChk, "wrap");
.
.
.
pack();
```

2. Функционалност

В Таблица 4 са представени всички функции, поддържани от диалоговия интерфейс.

Таблица 4

Функция	Описание
„Избор на мотор”	Изборът на мотор за управление се извършва от падащото меню в горния ляв ъгъл на диалоговия прозорец. Броят на моторите е ограничен до 8, поради спецификация на контролера. Моторите са номерирани от 0 до7. На номера на всеки мотор съответства определен порт на контролера.
„Включен”	Тази функция включва и изключва избрания мотор. При промяна на позиция или ускорение на изключен мотор, промените няма да се отразят до включването на мотора.
„Степер”	Степерът е функция, позволяваща плавно преминаване от текуща към нова позиция на избрания мотор.
„Спрян”	Отразява моментното състояние на избрания мотор.
„Задай позиция”	Функция, променяща текущата позиция на избрания мотор. Позицията се измерва в градуси и може да варира между 0° и 180°. Промяната се извършва чрез плъзгач. Тази функция може да се използва в комбинаци със „Степер”.
„Ускорение”	Функция, променяща ускорението на избрания мотор. Промяната се извършва чрез плъзгач.
„Максимална” и „Минимална позиция”	Функции, определящи интервал на движение за избрания мотор. При стартиране на приложението този интервал е от 0° до 180° и може да се коригира чрез плъзгач.

ЗАКЛЮЧЕНИЕ

Настоящата работа има преди всичко за задача да представи един опит за управление на роботизиран хардуер, посредством базирана на Java технология за човеко-машинен интерфейс. В този свой вариант работата подсказва и показва преди всичко, възможности за компютърно управление на роботизиран агрегат за работа в трудно-достъпни или опасни условия. Тази разработка може да бъде обогатена с Java-модули, имплементирани на базата на изкуствен интелект, за реализиране на проблемно-ориентирани решения - адекватни на конкретните реални ситуации, в рамките на зададен проблем.

ЛИТЕРАТУРА

[1] MiG Layout Quick Start Guide. Oct. 2010 http://www.miglayout.com/QuickStart.pdf
[2] Preston S. The Definitive Guide to Building Java Robots. Apress, 2006

За контакти:

Калоян Миронов, Русенски университет “Ангел Кънчев”, тел.: 0887-187- 149, e-mail: mr.kaloyan@gmail.com

Докладът е рецензиран.

РУСЕНСКИ УНИВЕРСИТЕТ „АНГЕЛ КЪНЧЕВ"
UNIVERSITY OF RUSE „ANGEL KANCHEV"

Д И П Л О М А

Програмният комитет на
Научната конференция РУ&СУ'10
награждава с КРИСТАЛЕН ПРИЗ
"THE BEST PAPER"
КАЛОЯН МИРОНОВ
автор на доклада
"Диалогов интерфейс за управление на роботизирана ръка"

D I P L O M A

The Programme Committee of
the Scientific Conference RU&SU'10
Awards the Crystal Prize
"THE BEST PAPER"
to KALOYAN MIRONOV
author of the paper
"Dialog Interface for Controlling a Robotic Arm"

РЕКТОР
RECTOR

доц. д-р Христо Белоев
Prof. D-r Hristo Beloev

01.11.2010