

Улази-излаз из корисничког простора

верзија 1.0

Одсек РТ-РК

9. децембар 2025.

1 Увод

У овој вежби биће приказано како се на Линукс оперативном систему из корисничког простора управља улаз-излаз опште намене (енгл. GPIO). GPIO омогућује да било која ножица (енгл. pin) чипа на RPi платформи се може иконфигурисати да ли је улаз или излаз, прочитати или уписати логичка вредност са њега. То се постиже извршавањем датотечких операција (`open()`, `read()`, `write()`) над датотеком `/dev/gpio_stream` GPIO руковаоца (енгл. Driver). Помоћу GPIO руковаоца се рукује DC мотором брисача (енгл. Wiper motor) и корачним мотором (енгл. Stepper motor) уписом логичке 0 или 1 на одређене ножице што ће резултовати напонским сигнаlima 0 V односно 3.3 V.

2 Радни процес

С обзиром да постоји ограничен број поставки (енгл. Setup), као и лоше искуство са пребацивањем хардвера са једног RPi на други, постојаће један RPi са повезаном поставком на којем ћете извршавати своје програме, које сте искомпајлирали на личном RPi. Подсетимо се типичних корака ради провере рада мреже и интернета на RPi:

```
1 ifconfig
2 ping 8.8.8.8
3 ping google.com
```

Проверимо да ли је мрежа проходна према поставци:

```
1 ping rpi-setup-1.local
```

Повежемо се на поставку:

```
1 ssh pi@rpi-setup-1.local
```

Лабораторијска вежба се може отпаковати или клонирати са

```
1 git clone https://github.com/MilosSubotic/OSuRV_Labs
```

и позиционирамо се у њу:

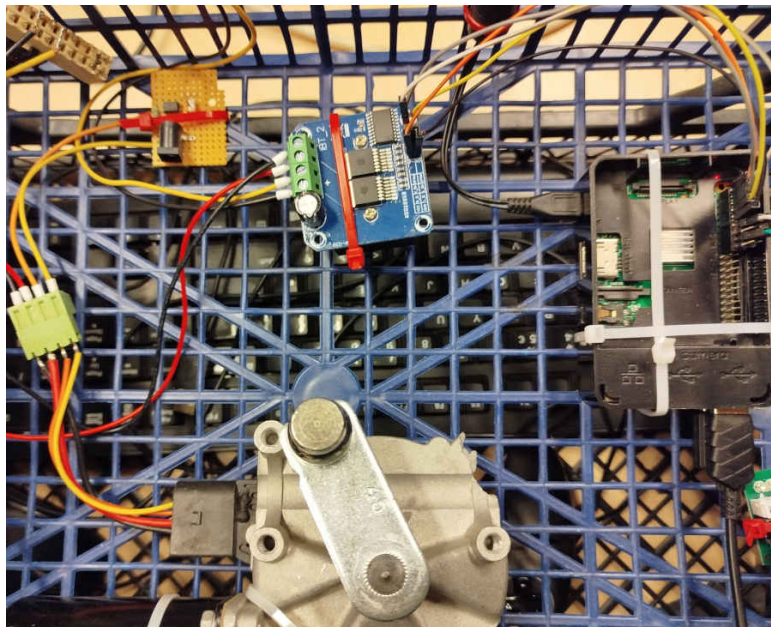
```
1 cd OSuRV_Labs/L5_User_Space/
```

Један судент или асистент треба да иницијално покрене драјвер на поставци:

```
1 cd SW/Driver/gpio_ctrl/  
2 make  
3 make start  
4 dmesg
```

3 DC мотор

На следећој слици је приказана хардверска поставка са DC мотором брисача.



Слика 1: Хардверска поставка са DC мотором брисача

На локалном RPi искомпајлирати следећи тестни програм.

```
1 cd SW/Test/test_gpio/  
2 ./waf configure  
3 ./waf build
```

Копирати искомпајлиран програм на setup RPi

```
1 scp ./build/test_gpio pi@rpi-setup-1.local:/home/pi/Public
```

На setup RPi преко ssh везе извршити:

```
1 cd /home/pi/Public  
2 ./test_gpio w 2 0  
3 ./test_gpio w 3 0  
4 ./test_gpio w 4 1  
5 ./test_gpio w 2 1
```

Мотор би требао да се окреће. За окретање у другу страну:

```
1 ./test_gpio w 2 0  
2 ./test_gpio w 3 1  
3 ./test_gpio w 4 0  
4 ./test_gpio w 2 1
```

за заустављање:

```
1 ./test_gpio w 2 0
```

Са асистентима искоментаришите шта се десило и прођите кроз код `test_gpio.c` од 130–150 линије. `open()` отвара GPIO руковаоца. Припрема се пакет од 3 бајта за писање на GPIO са следећим вредностима:

1. 'w' за упис
2. број пина
3. вредност (0 или 1)

Позива се `write()` функција са уписом садржаја пакета и провером успешности.

4 Joypad

USB Joypad се тестира на setup RPi:

```
1 lsusb
2 ls -l /dev/input/js0
3 groups
4 jstest /dev/input/js0
```

Компајлирање у локалу:

```
1 cd SW/Test/test_joypad/
2 ./waf configure
3 ./waf build
4 scp ./build/test_joypad pi@rpi-setup-1.local:/home/pi/Public
```

Преко ssh тестирати:

```
1 ./test_joypad
```

Притискати дугмиће. Проћи кроз код. Новитет је коришћење `read()` функције. Joypad има предефинисан `struct js_event` као пакет за читање статуса дугмића и joystick оса.

Ако има времена пробати апликацију `SW/App/1_Joy_Wiper` која обједињује joypad и мотор и проћи кроз код. Искомпајлирати са:

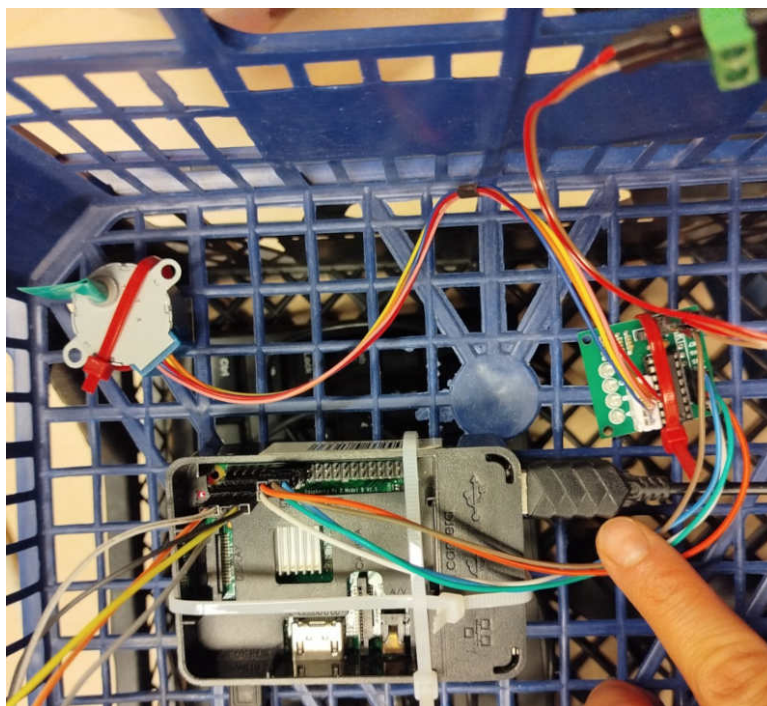
```
1 cd SW/App/1_Joy_Wiper/
2 ./waf configure
3 ./waf build
4 scp ./build/joy_wiper pi@rpi-setup-1.local:/home/pi/Public
```

Преко ssh тестирати:

```
1 ./test_joypad
```

5 Задатак: корачни мотор

На следећој слици је хардверска поставка са корачним моторима.



Слика 2: Хардверска поставка са корачним моторима

Пример за тестирање хардверске поставке мотора на setup RPi са:

```
1 cd SW/Test/test_gpio/  
2 ./waf configure  
3 ./waf build  
4 ./test_steppers.sh
```

Мотори се непрестано врте. Поиграти се са `STEP_DURATION`. Проћи кроз код скрипте и индуктивно закључити поступак корачања мотора.

Исто што `test_steppers.sh` скрипта ради реализовати у C апликацији `SW/App/4_Stepper` водећи се примером `test_gpio.c`.

Ако има времена, може се урадити и задатак `SW/App/5_Joy_Stepper` који спаја јоурад и корачни мотор, водећи се примером `SW/App/1_Joy_Wiper`.