|  |  |  |
| --- | --- | --- |
|  | **УНИВЕРЗИТЕТ У НОВОМ САДУ**  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА** |  |

|  |
| --- |
| **УНИВЕРЗИТЕТ У НОВОМ САДУ**  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  **НОВИ САД**  **Департман за рачунарство и аутоматику**  **Одсек за рачунарску технику и рачунарске комуникације**  **ИСПИТНИ РАД**  **Кандидати: Дамјан Гламочић, Михаило Марковић**  **Број индекса: РА65/2015, РА191/2015**  **Предмет: Системска програмска подршка у реалном времену II**  **Тема рада: Linux руковалац за Raspberry Pi 2 платформу за комуникацију корисничког програма и сензора HC-SR04**  **Ментор рада: Владимир Маринковић**  **Нови Сад, јануар, 2018.** |
|  |

Садржај

[1. Увод](#_Toc471814223) 3

[1.1 Задатак](#_Toc471814224) 3

[1.2 Анализа проблема](#_Toc471814225) 4

[2. Концепт решења](#_Toc471814226) 5

2.1 GPIO руковалац 5  
 2.2 Тест апликација 7

# 1. Увод

## 1.1 Задатак

Потребно је реализовати Linux руковалац за Raspberry Pi 2 уређај који треба да омогући комуникацију корисничког програма и ултразвучног сензора HC-SR04 који је повезан на GPIO пролазе доступне на уређају. Руковалац је потребно реализовати коришћењем прекида(interrupts).

## Руковалац треба да омогући:

## Генерисање Trigger (Ping) сигнала на захтев корисничке апликације

## Одређивање ширине импулсног Echo сигнала у прекидној рутини (време протекло између растуће и опадајуће ивице). Ширина Echo сигнала је пропорционална удаљености препреке.

## Преузимање измереног времена које одговара ширини импулса Echo сигнала, на захтев корисничке апликације.

У сврху провере функционалности развијеног руковаоца потребно је развити корисничку апликацију за детекцију препрека. Апликација треба обрађује и исписује информације пристигле са сензора. Обрада подразумева да апликација преко уређаја тј. руковаоца преузима подаке са сензора, рачуна удаљеност препреке и филтрира добијене резултате високопропусним IIR филтром.

## 1.2 Анализа проблема

Главни проблем приликом имплементације руковаоца представља реализација слања и пријема ултразвучног сигнала на периферију. За потребу пријема сигнала мора се написати рутина за обраду прекида (енг. *i****nterrupt service routine***), док се емитовање сигнала мора обављати у ***write***функцији руковаоца.  
 Руковалац ће, осим прекида, користити и тајмере високе прецизности, зато што ће временски интервал са којим ћемо радити бити на нивоу милисекунди.  
 Рачунање раздаљине периферије и објекта се рачуна по TRD формули која гласи: D = C\*T, где је C константа брзине звука у ваздуху, а Т је временски интервал у коме је периферни уређај емитовао и примио рефлектовани сигнал. У нашем случају Т ћемо поделити са 2 јер нам је потребно само време за које сигнал стигне до објекта.  
 У руковаоцу је такође потребно написати и IIR функцију која ће реализовати филтрирање резултата високопропусним IIR филтром, а филтрирање се изводи по формули y[i] := α \* (y[i-1] + x[i] - x[i-1]).

Апликација кориснику мора понудити операцију за почетак емитовања поворке импулса (шаље се 8 ултразвучних таласа фреквенције 40KHz) и тада се активира тајмер. Када уређај детектује рефлектовани сигнал, тајмер се зауставља. Тестна апликација затим треба да обради и испише информације пристигле са сензора.

Поред самог руковаоца, потребно је написати и тест апликацију која ће радити са имплементираним фукцијама руковаоца, обрађивати пристигле податке и приказивати их кориснику.

Са хардверске стране, потребно је повезати пинове Raspberry Pi 2 платформе са пиновима ултразвучног сензора HC-SR04 на одговарајући начин (конектором правилно повезати пинове сензора 5V, GND, као и 2 GPIO пина за пренос података са њима одговарајућим пиновима на Raspberry Pi 2 платформи).

# 2. Концепт решења

**2.1. GPIO руковалац**

Да би се омогућило правилно функционисање руковаоца, постављени су одговарајући смерови за пинове, и то Trigger пин као output, а Echo пин као input. То се врши у оквиру иницијализујуће функције ***gpio\_driver\_init*** помоћу функција ***SetGpioPinDirection*** и ***SetInternalPullUpDown***. Помоћу функције ***register\_chrdev*** региструје се уређај. U овиру иницијализујуће функције извршена је иницијализација прекида и придруживање interrupt handler функције. Такође, у оквиру ове функције извршена је и иницијализација бафера података у који ће се смештати подаци са руковаоца као и ремапирање GPIO регистарског простора са физичког на виртуелни адресни простор.

Функција ***gpio\_driver\_read*** обавља највећи део посла руковаоца. На самом почетку њен *trigger* пин се сетује на 1, а након 10 микросекунди враћа на 0. Та операција сензору представља информацију да почне са емитовањем сигнала. Након што сигнал почне са емитовањем, покреће се тајмер високе резолуције који бележи време почетка емитовања сигнала. Функција затим чека промену променљиве *completed* да би наставила са даљим радом, а њу мења сам *interrupt* када заврши своје извршавање. Када добије информацију да је прекид извршен, функција уписује трајање емитовања ултразвучног сигнала у *gpio\_driver\_buffer*, а затим се тај бафер уписује у посебну датотеку из које ће касније читати тест апликација.

Функција прекида ***h\_irq\_gpio\_echo\_falling\_edge*** реализује прекид када се на *echo* пину појави falling edge сигнала. Функција бележи време када се активирао прекид, а затим рачуна трајање емитовања ултразвучног сигнала као разлику времена завршетка из саме функције прекида и забележеног времена почетка из ***gpio\_driver\_read*** функције, а онда поставља flag *completed* на 1 како би ***gpio\_driver\_read*** могла да настави са радом.

По угледу на иницијализујућу функцију потребно је имплементирати и деиницијализујућу, ***gpio\_driver\_exit*** у којој ће се ослободити сви заузети пинови, онемогућити прекиди и ослободити заузети простор за бафере, као и одмапирати GPIO физички адресни простор са виртуелног. На самом крају је потребно и избрисати руковалац из регистра позивањем функције ***unregister\_chrdev***.

**2.2 Тест апликација**

Главни задатак тест апликације је да преузима податке које је проследио руковалац, обради их и прикаже кориснику. То ради тако што чека да корисник унесе команду за почетак емитовања сигнала. Када се покрене, апликација потребне променљиве за ***main*** и ***IIR*** функције, а потом отвара датотеку у коју руковалац уписује податке и чита податке из ње.   
Подаци се затим обрађују у реалном времену тако да се време емитовања сигнала, које је прослеђено из руковаоца, конвертује помоћу дате формуле у меру удаљености објекта од мерног уређаја. Подаци се потом филтрирају помоћу имплементираног ***IIR*** филтра и, коначно, исписују крајњем кориснику у терминал.