

**School of Engineering Technology**

**Main Campus, Off Hennur-Bagalur Main Road, Chagalahatti, Bengaluru-562149**

***A***

***DISSERTATION REPORT ON***

***“QUICK REACTION GAME”***

Submitted to

***CMR University School Of Engineering Technology, Bagalur***

for the partial fulfillment of the Requirement for the Award of the Degree of

***B.TECH***

***IN***

***COMPUTER SCIENCE AND ENGINEERING***

Submitted by:

***T.RAKSHA.BOPANNA(18BBTCS132)***

***THILOTHY.P(18BBTCS134)***

***VENKATBHARAT.P(18BBTCS144)***

***VISHNU.DEEPAK(18BBTCS149)***

***Under the Guidance Of***

***Prof. PALLAVI.M***

**DEPARTMENT OF**

**COMPUTER SCIENCE AND ENGINEERING CMR UNIVERSITY BAGALURE**

**2018-19**



**School of Engineering Technology**

**Main Campus, Off Hennur-Bagalur Main Road, Chagalahatti, Bengaluru-562149**

**DEPARTMENT OF** **COMPUTER SCIENCE AND ENGINEERING**

***CERTIFICATE***

*Certified that the project work entitled* ***QUICK REACTION GAME*** *carried out by Mr./Ms.* ***T.RAKSHA.BOPANNA(18BBTCS132), THILOTHY.P(18BBTCS134), VENKATBHARAT(18BBTCS144), VISHNU.DEEPAK(18BBTCS149)*** *in partial fulfillment for the award of Bachelor of Engineering / Bachelor of Technology in* ***COMPUTER SCIENCE AND ENGINEERING*** *of the CMR University, Bagalur during the year 2018-19. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.*

***Name of the Guide : PALLAVI.M Name of the Dean:Dr.HARISH.RAMANNA***

*Signature of the Guide*   *Signature of the Dean*

**External Viva**

**Name of the examiners Signature with date  
1  
2**

***DECLARATION***

***We,T.RAKSHA.BOPANNA,THILOTHY.P,VENKATBHARAT.P,VISHNU.DEEPAK*** *students of CMR university school of engineering and technology, bagalur hearby declare that the dissertation entitled* ***“ QUICK REACTION GAME”*** *embodies the report of our project carried out independently by us during first semester of* ***B.TECH in computer science and engineering,*** *under the supervision and guidance of* ***Prof. PALLAVI.M*** *Department of Computer Science and Engineering and this work has been submitted for the partial fulfillment of the requirements for the award of the B.Tech degree.*

*We have not submitted the matter embodies to any other university or institution for the award of other degree.*

*Date :*

*Place :*

**T.RAKSHA.BOPANNA(18BBTCS132)**

**THILOTHY.P(18BBTCS134)**

**VENKATBHARAT.P(18BBTCS144)**

**VISHNU.DEEPAK(18BBTCS149)**

***ACKNOWLEDGEMENT***

*We express our foremost gratitude to* ***Dr.HARISH.RAMANNA****, Dean for his constant support.*

*We express our foremost gratitude to our guide* ***Prof.PALLAVI.M****, Dept of*

*Computer Science & Engineering, CMR University, Bagalur for her inspiration, adroit guidance, constant supervision, direction and*

*discussions in successful completion of this dissertation.*

*Our sincere thanks to all teaching and non-Teaching Staff of* ***Computer Science***

***& Engineering Department*** *for all the facilities provided, without which, we could not*

*have progressed with our work. Thanks to our parents who have been a great source of*

*strength in the completion of this dissertation.*

**T.RAKSHA.BOPANNA(18BBTCS132)**

**THILOTHY.P(18BBTCS134)**

**VENKATBHARAT.P(18BBTCS144)**

**VISHNU.DEEPAK(18BBTCS149)**

**ABSTRACT**

This project gives you the opportunity to use electronics to create a quick reaction game which you will program using Python.

## **What you will learn**

By making this quick reaction game, you will learn:

* How to wire a simple circuit that includes a breadboard, LED, resistor, wires, and buttons
* How to write a program to control the circuit
* How to use variables to store information
* How to get user information like a player’s name and use it in the game.

This resource covers elements from the following strands of the [*Raspberry Pi Digital Making Curriculum*](https://www.raspberrypi.org/curriculum/)*:*

* [*Use basic programming constructs to create simple programs*](https://www.raspberrypi.org/curriculum/programming/creator)
* [*Combine inputs and/or outputs to create projects or solve a proble**m*](https://www.raspberrypi.org/curriculum/physical-computing/builder)

**TABLE OF CONTENTS**

#### TITLE PAGE NO

DECLARATION …………………………………………………….…….…………... i

ACKNOWLEDGEMENT ……………………………………………….…….…………..ii

ABSTRACT …………………………………………………………….……………...….iii

TABLE OF CONTENT …………………………………………………..………………..iv

LIST OF FIGURES …………………………………………………………..………….....vi

CHAPTER

1. PREAMBLE .……………………………………………1

Introduction …………………………….…………#

Literature Survey……………………………………#

Problem Statement ………………………………...……#

Objective ………………………………...……#

Methodology ………………………………...……#

1. GENERAL ASPECTS AND TECHNOLOGY ……………………………..……#

1. IMPLEMENTATION .……………………………………...…#

4. RESULTS AND DISCUSSIONS .……………………………………...…#

CONCLUSIONS ………………………………………………………………………..…….#

REFERENCES ………………………………………………………………………...#

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| FIGURE NUMBER | NAME OF THE FIGURE | PAGE.NO |
| 1.1 | CIRCUIT DIAGRAM | 11 |

**CHAPTER 1**

**PREAMBLE**

**1.1 Introduction**

This project gives you the opportunity to use electronics to create a quick reaction game which you will program using Python. If you have little or no experience of creating circuits, don’t worry: this guide will walk you through it and by the end you will have a fun game to play with your friends.

In this resource, you’re going to make a quick reaction game using a few electronic components and a Python script. If you’ve never before used a breadboard, some buttons, and an LED, you might find it helpful to work through some of the exercises in [**Physical Computing with Python**](https://projects.raspberrypi.org/en/projects/physical-computing) first. This will give you a better understanding of how to control components with the Raspberry Pi’s GPIO pins.

This is the circuit you are going to build, consisting of two push-to-make buttons and an LED.

## Controlling the light

When programming, it makes sense to tackle one problem at a time. This makes it easier to test your project at various stages.

**1.2 Literature Survey**

**We found this project on the raspberry pi projects website.**

**Raspberry** is a development board that has made a lot of noise since its introduction to the public. The cost and the wide support community has further added to the wide range of applications this board supports.

**1.3 Problem statement**

**1.4** **Objective**

By making this quick reaction game, you will learn:

* How to wire a simple circuit that includes a breadboard, LED, resistor, wires, and buttons
* How to write a program to control the circuit
* How to use variables to store information
* How to get user information like a player’s name and use it in the game.

This resource covers elements from the following strands of the[*Raspberry Pi Digital Making Curriculum*](https://www.raspberrypi.org/curriculum/)*:*

**1.5 Methodology**

* Building the circuit
* Controlling the light

## Adding an element of surprise

## Detecting the buttons

## Get player names

* Looping the game
* Adding scores

**CHAPTER 2**

**GENERAL ASPECTS AND TECHNOLOGY**

**The technology used is raspberry pi and python3**

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

What’s more, the Raspberry Pi has the ability to interact with the outside world, and has been used in a wide array of digital maker projects, from music machines and parent detectors to weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work

**Python** is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language),high level programming language for [general-purpose programming](https://en.wikipedia.org/wiki/General-purpose_programming_language). Created by [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) and first released in 1991, Python has a design philosophy that emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability), notably using [significant whitespace](https://en.wikipedia.org/wiki/Significant_whitespace). It provides constructs that enable clear programming on both small and large scales. In July 2018, Van Rossum stepped down as the leader in the language community.

Python features a dynamic type system and automatic [memory management](https://en.wikipedia.org/wiki/Memory_management). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](https://en.wikipedia.org/wiki/Imperative_programming), f[uncti](https://en.wikipedia.org/wiki/Functional_programming)onal and [procedural](https://en.wikipedia.org/wiki/Procedural_programming), and has a large and comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library).[[29](https://en.wikipedia.org/wiki/Python_(programming_language)" \l "cite_note-About-29)]

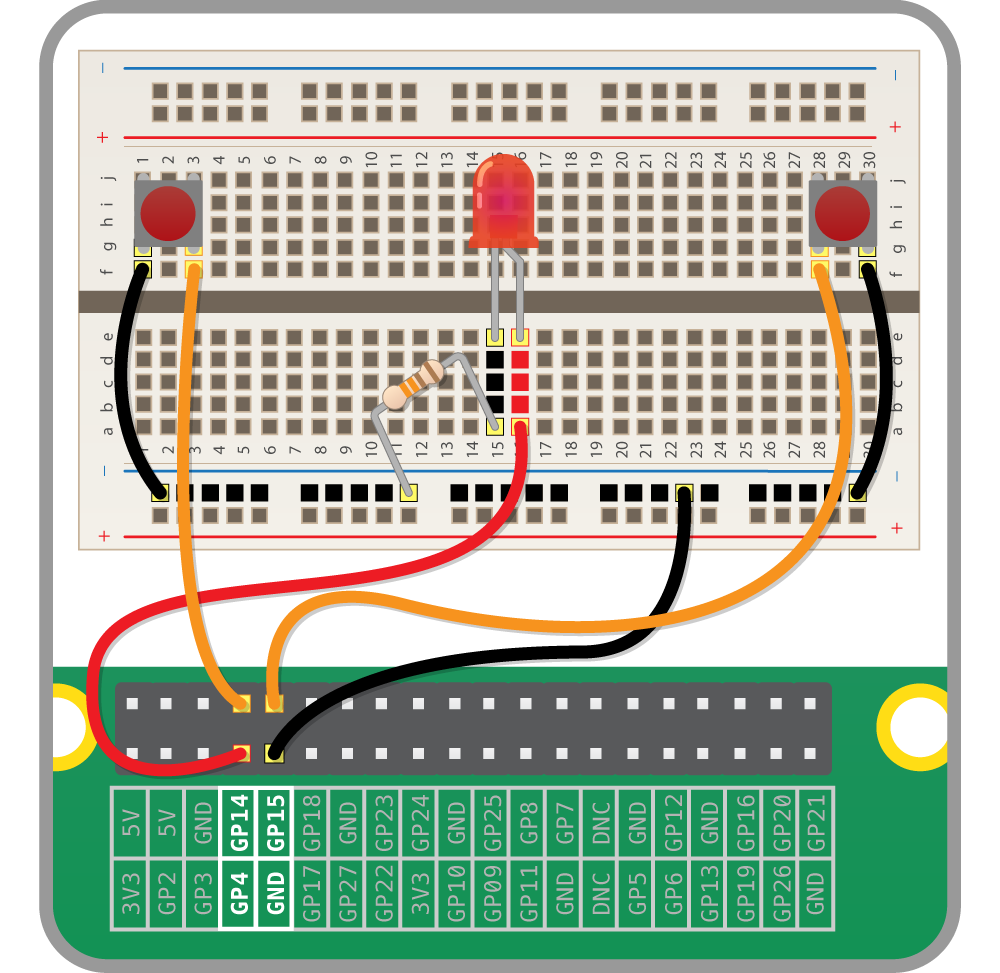
Python interpreters are available for many [operating systems](https://en.wikipedia.org/wiki/Operating_system). [CPython](https://en.wikipedia.org/wiki/CPython), the referenceimplementation of Python, is [open sou](https://en.wikipedia.org/wiki/Open-source_software)rce software and has a community-based development model, as do nearly all of Python's other implementations. Python and CPython are managed by the non-profit python software foundation.

**CHAPTER 3**

**IMPLEMENTATION**

The first step is to build the circuit.

## **Building the circuit**

This is the circuit we are going to build, consisting of two push-to-make buttons and an LED.

* Take one of your tactile buttons and push it into the holes on your breadboard, with one set of legs on row **H** and one set of legs on row **J**.
* Repeat the last step with the second button, placing it at the other end of the breadboard on the same row.
* Place an LED with the longer leg above the ridge in the breadboard in **D16** and the shorter leg in **D15**. The numbering will depend on your breadboard so make sure that you check the diagram below.
* Next push one leg of the resistor into the same column (**15**) as the short leg of the resistor and the other leg into a hole along the blue strip.
* Now it’s time to add the jumper wires. Start by taking two male-to-male jumper wires and placing one end in a hole next to the outside leg of the left hand button, and the other end in a hole along the blue strip. Repeat this step with the right hand button.
* Then with a male-to-female jumper wire, connect **GPIO14** to a hole on the breadboard in line with the other leg of the left hand button. Repeat this step for the right hand button, only this time connecting it to **GPIO15**.
* Using another male-to-female jumper wire, connect **GPIO4** to a hole on the breadboard in line with the long leg of the LED.
* Finally, connect a **GND** GPIO pin to the blue strip on the breadboard with the remaining male-to-female jumper wire.

Writing the code:

First you will need to import the modules and libraries needed to control the GPIO pins on the Raspberry Pi. Type:

* from gpiozero import LED,Button
* from time import sleep

As you are outputting to an LED, you need to set up the pin that the LED connects to on the Raspberry Pi as an output. First use a variable to name the pin and then set the output:

* led=LED(4)

Next add a line to turn the LED on:

* led.on()

Now add a line to wait 5 seconds by typing:

* sleep(5)

Then add a line to turn the LED off like this:

* led.off()

Here,uniform allows for the random selection of a decimal (floating point) number from a range of numbers.

* from random import uniform

Then locate the line sleep() and amend it so that it reads:

* sleep(uniform(5, 10))

Intitialize buttons

* led = LED(4)
* right\_button=Button(15)
* left\_button=Button(14)

Then underneath led.off() you can add a function that will be called whenever a button is pressed, which will tell you which pin the button was on:

* def pressed(button):
* print(str(button.pin.number)+’won the game’)

To finish off, when either button is pressed, the function will be called. If the right button is pressed, then you can send the string ‘right’ to the pressed function. If the left\_button is pressed, then you can send the string’left’.

* right\_button.when\_pressed = pressed
* left\_button.when\_pressed = pressed

**CHAPTER 4**

**RESULTS AND DISCUSSIONS**

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries.

**CONCLUSION**

**REFERENCES**

|  |
| --- |
| 1.) https://projects.raspberrypi.org/en/projects/python-quick-reaction-game/5 |