GROUP NUMBER:

ROLL NUMBER 1: 200260040 NAME 1: Rhishabh Suneeth
ROLL NUMBER 2: 200260048 NAME 2: Shadab Anjum

TITLE: <FINAL, AS BUILT> DINO GAME USING ARDUINO

< 1 line title >
[note: the final project as built may be quite different from your initial proposal. *Everything in this report must refer to the final project as presented in the demo.*]

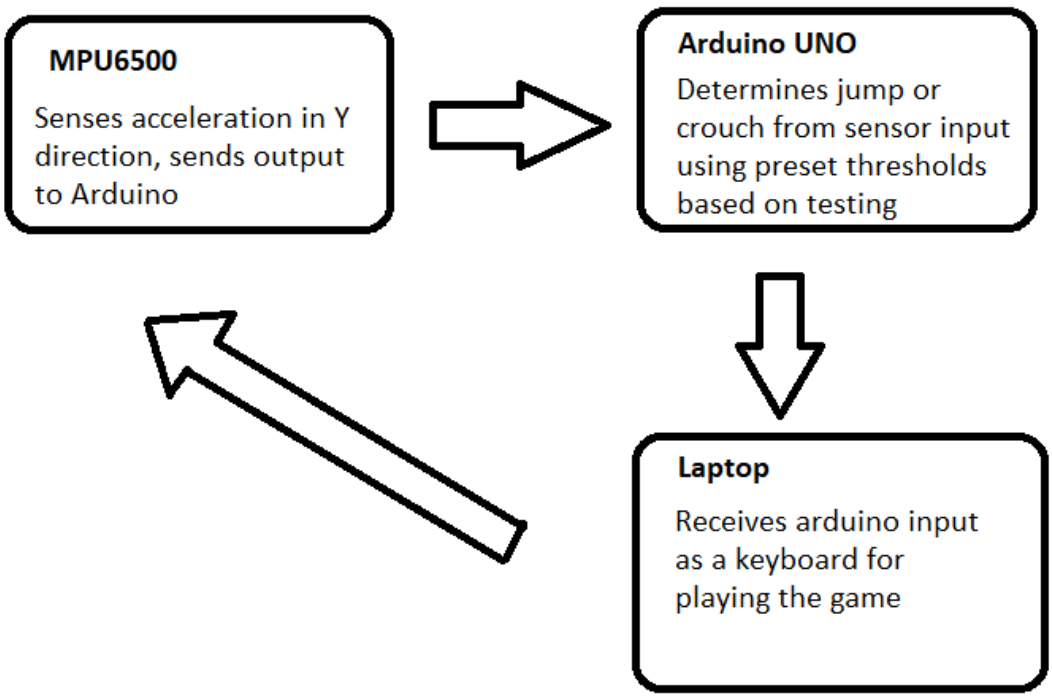
ABSTRACT:

< describe your project idea in 5 lines or less >

In this project, we use the Arduino along with sensors to detect motion (jump/up/crouch/down) to enable users to play the dino game on their laptop(windows) in an interactive manner.

PROJECT DETAILS:

< Give a detailed description of your project idea >
You must provide a block diagram of the major components of the project.
[1]



Use blocks to specify all sensors/input/output elements used. Give the details of the parts used, especially if they are commercially purchased parts (like ultrasonic sensors, LCD screens etc)
[2]
External bread-boarded circuits should be discrete blocks – give the circuit diagram corresponding to those circuits as sub-figures. Hand-drawn circuit diagrams are *not* acceptable – use a proper software like circuit lab or ltspice etc you have learned in earlier labs.

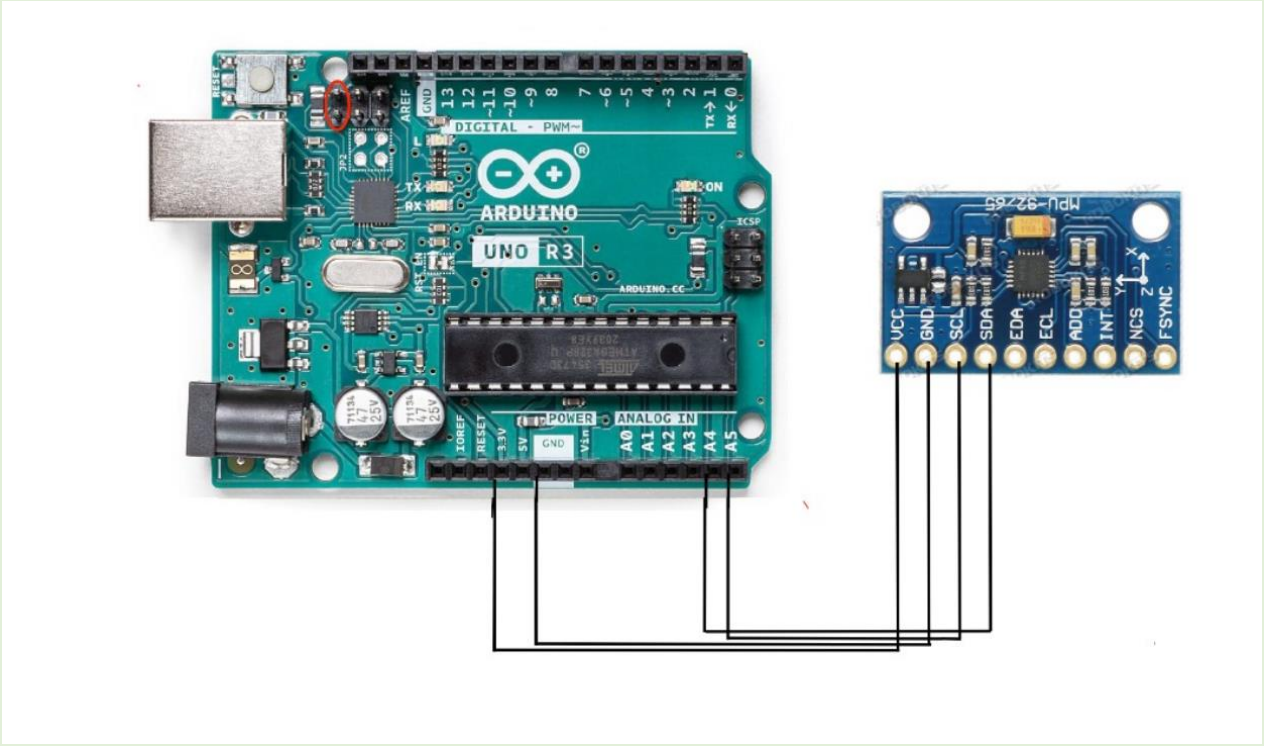
ARDUINO TO SENSOR CONNECTIONS:

V=3.3V

GND

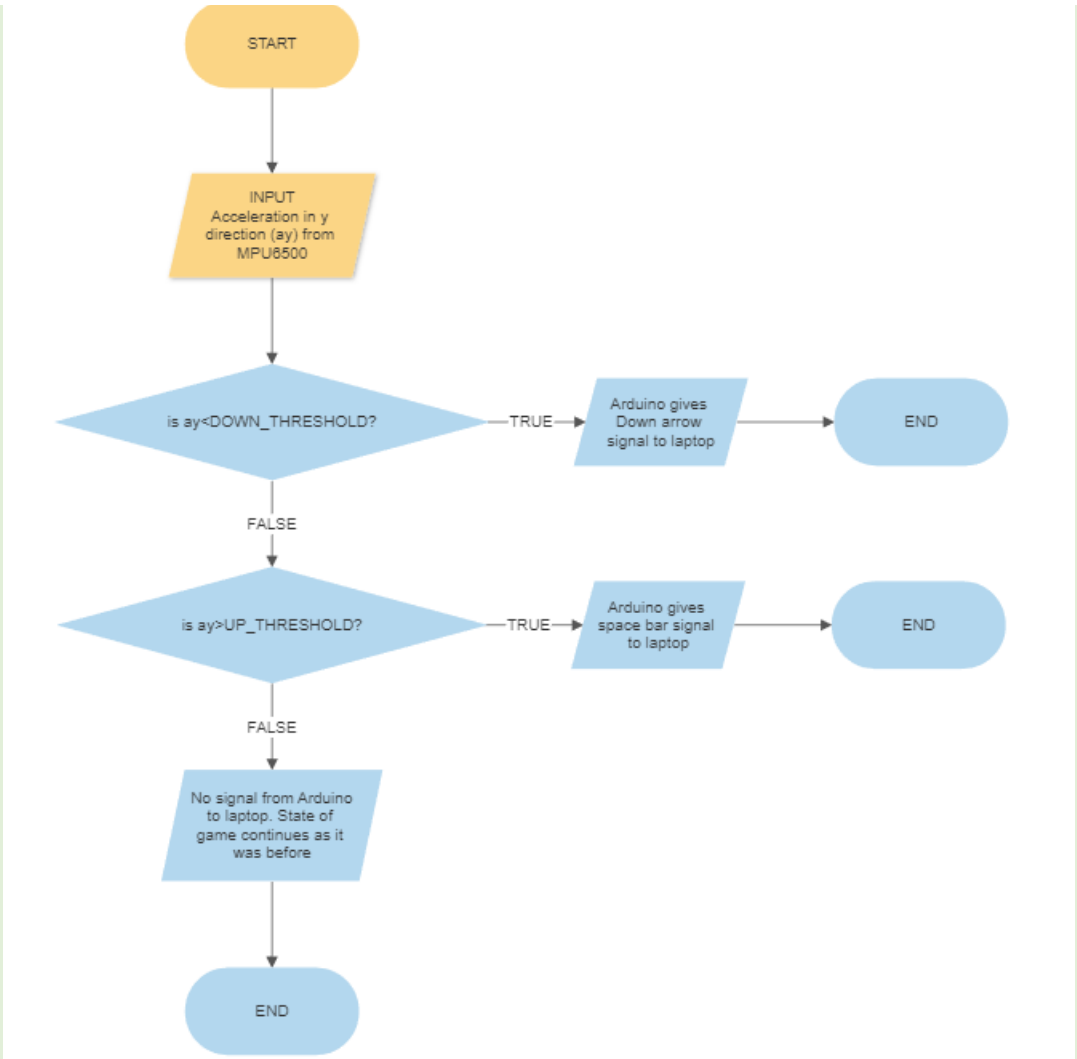
SCL connected to A5

SDA connected to A4



The red circle denotes the 2 ICSP pins used to reset the USB connection, so that we can upload the corresponding keyboard hex codes.

[3]
The Arduino with the required algorithm should be in a separate block.
The **algorithm flowchart** must be shown here. Program code is to be included as an appendix to the report (see below for format)



Mark on your block diagram which group member was responsible majorly for working on which block. Writing 'both did everything' is not acceptable – surely you must have shared workload among the group members. The TA's have been tracking your progress.

Calibration, threshold measurement and sensor testing: Shadab

Integrating Arduino with sensor and laptop: Rhishabh

[4]
If your project has significant analog circuits as part of the design, include LTSpice simulation circuit diagrams and simulation result plots of the analog component.

Not applicable

MAIN COMPONENTS NEEDED TO BUILD THE PROJECT:

Give an inventory of all the components you used to complete the project. If you needed to purchase some components other than the ones provided in your kit, please mention them separately.

Arduino UNO, laptop (preferably windows), MPU6500 accelerometer and gyroscope, cardboard, and jumper wires.

RESULTS:

Summarize the results of your project.

Provide photos or links to video recording of your working project output. In the project demo and viva we expect a fully working end-result of your project work. But sometimes a last minute part failure may cause problems. In that case, we would like to see that your project worked at *some* time!

Plus, a photogenic project output gets you a chance to get on the 'Projects Hall-of-Fame' poster board.

Successfully implemented Arduino as a keyboard using FLIP software and HEX codes for setting up Arduino as an HID device. Using HID codes for windows we mapped the keycodes to respective keys and passed keyboard input in a serial manner using a keyboard report buffer. The Arduino detects sensor inputs and uses thresholds to determine whether the person has jumped or crouched and gives the corresponding keyboard input to the keyboard which enables a person to play the game. The Arduino along with the sensor is setup in a cardboard box, which is bound to the chest using a belt giving the user an interactive gaming experience with the Dino game.

Demos: https://drive.google.com/drive/folders/18V33vhrKzbMQonkDE428TpwLzEjgqPko?usp=sharing

APPENDIX:

Program code is to be put here in the following fixed width font. Note that the code must be well commented and self-explanatory. The following is a shining example of well-written code:
(the color coding of functions is just a cosmetic add-on for readability)

```
#include<Wire.h>

const int mpu_addr=0x68;

uint8_t buf[8] = { 0 }; //Keyboard report buffer

#define PIN_U 4 // Pin for w


#define PIN_D 5 // Pin for a


#define UP_THRESHOLD 30000


#define DOWN_THRESHOLD 0


#define SPACE 44 //change this parameter based on keyboard HID value for SPACEBAR


#define DOWN 81 //change this parameter based on keyboard HID value for DOWN ARROW


void setup() {

Serial.begin(9600); // Setup Serial communication

//Set pinmode of Input pins

pinMode(PIN_U, INPUT); // for testing purpose

pinMode(PIN_D, INPUT); // for testing purpose

//MPU6500 setup for wire transmission

  Wire.begin();

  Wire.setClock(400000UL); // Set I2C frequency to 400kHz

  Wire.beginTransmission(mpu_addr);

  Wire.write(0x6B);

  Wire.write(0); // wake up the mpu6050

  Wire.endTransmission(true);

  Serial.begin(9600);
```

```
}
void loop() {
//When button representing SPACEBAR is pressed
double ay=Acc_Y();
if(ay>UP_THRESHOLD){
  releaseKey();
  buf[2] = SPACE; // UP keycode
  Serial.write(buf, 8); // Send keypress
  delay(300);
  releaseKey();
}
//code for testing Arduino as a keyboard using switch commented out after testing (SPACEBAR)
/*if (digitalRead(PIN_U) == HIGH) {
buf[2] = 44; // UP keycode
Serial.write(buf, 8); // Send keypress
delay(100);
releaseKey();
}
*/
//When button representing DOWN is pressed
if(ay<DOWN_THRESHOLD){
  releaseKey();
  buf[2] = DOWN; // DOWN keycode
  Serial.write(buf, 8); // Send keypress
  delay(500);
  releaseKey();
}
//code for testing Arduino as a keyboard using switch commented out after testing (DOWN ARROW)
/*if (digitalRead(PIN_D) == HIGH) { buf[2] = 81; // DOWN keycode
Serial.write(buf, 8); // Send keypress
}
if(digitalRead(PIN_D)== LOW){
  releaseKey();
}
}
*/
}
// Function for Key Release
void releaseKey() {

buf[0] = 0;
buf[2] = 0;
Serial.write(buf, 8); // Send Release key
}
```

```
// Function to get y acceleration
double Acc_Y()
{
  Wire.beginTransmission(mpu_addr);
  Wire.write(0x3D);
  Wire.endTransmission(false);
  Wire.requestFrom(mpu_addr,2);
  double AccY=Wire.read()<<8|Wire.read();
  Wire.endTransmission(true);
  return AccY;
}
```