

Caixa de Schrödinger

Objetivo

O objetivo deste experimento é ilustrar, por meio de simulações e analogias, como o princípio da superposição quântica e o ato de observação influenciam o comportamento de partículas subatômicas.

Lista de Materiais

1	Placa de Arduino
10	Jumpes
2	Micro servo
1	Resistor 300Ω
1	Fotoresistor
1	Lâmpada
	Pedaços de madeira
	Fonte de energia

Montagem e execução do experimento

- I. Pegue 6 pedaços de madeira de 30 x 30 mm para fazer a caixa que comportará os componentes. (Figura 1-3)

Figura 1: Pedaços de madeira



Figura 2: Esquema da caixa

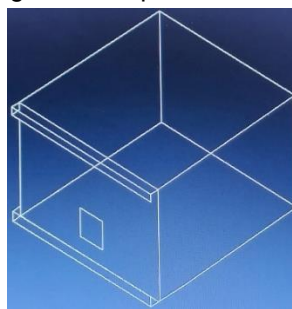


Figura 3: Caixa montada



Fonte: Valmir Santos (2024).

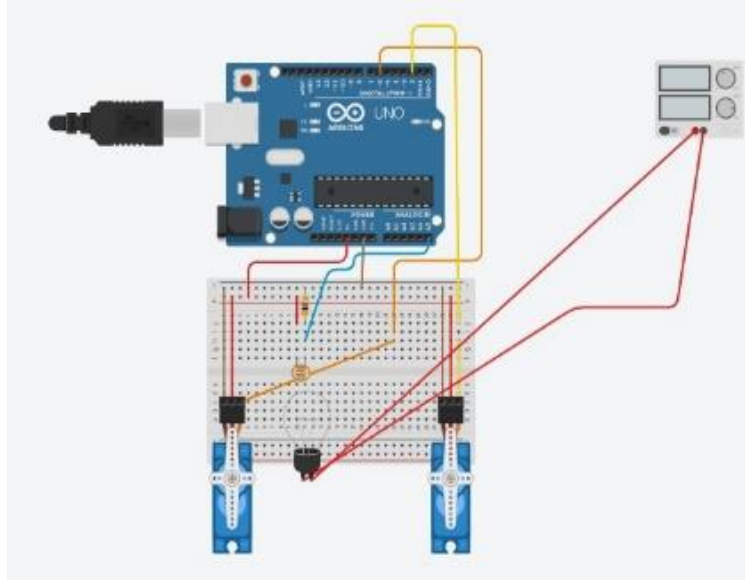
- II. Faça uma porta para manter a caixa fechada quando necessário (Figura 4)

Figura 4: Porta para a caixa



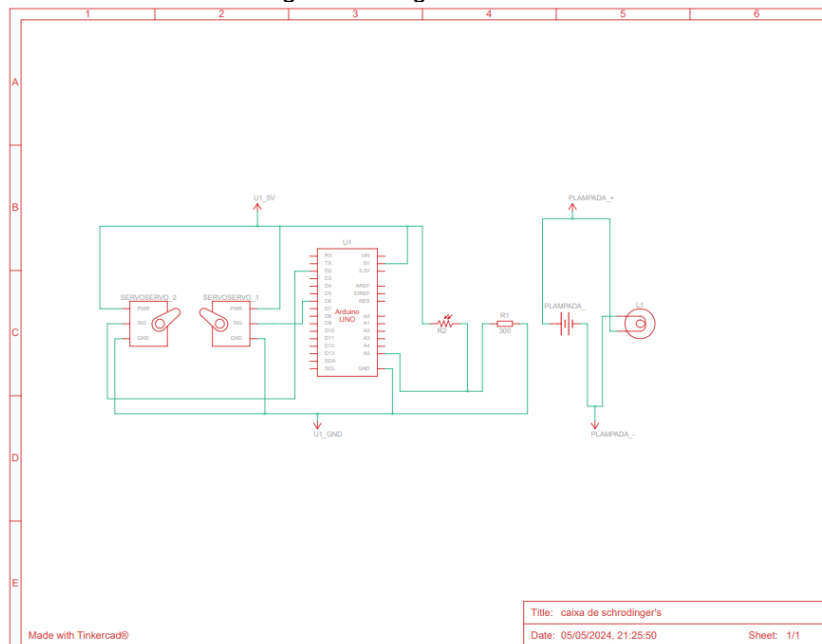
Fonte: Valmir Santos (2024)

- III. Esquema elétrico com os periféricos do experimento feito no Tinkercad (Figura 5) e diagrama elétrico (Figura 6)



Fonte: Valmir Santos (2024).

Figura 6: Diagrama Elétrico

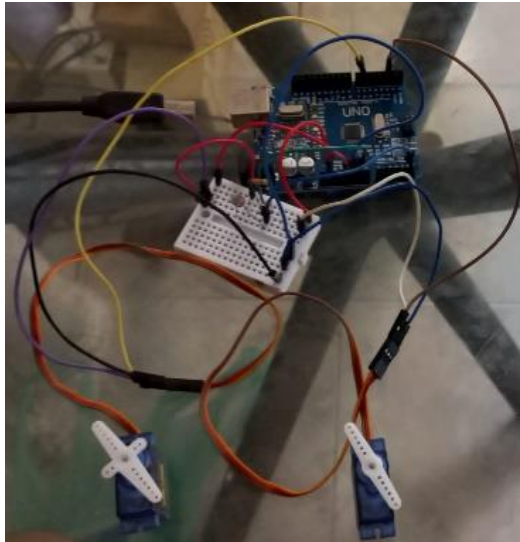


Fonte: Valmir Santos (2024)

- IV. Montagem experimental fora da caixa e dentro da caixa



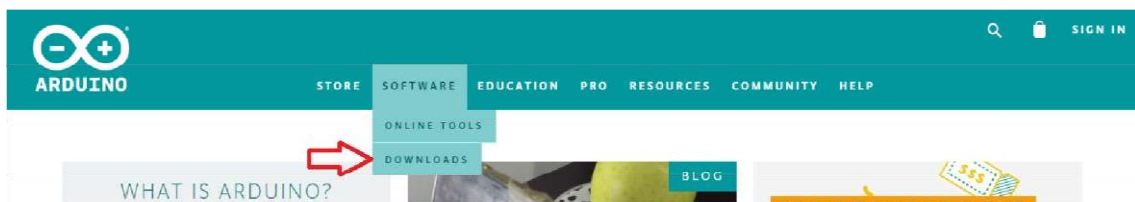
GET UP
SCIENCE



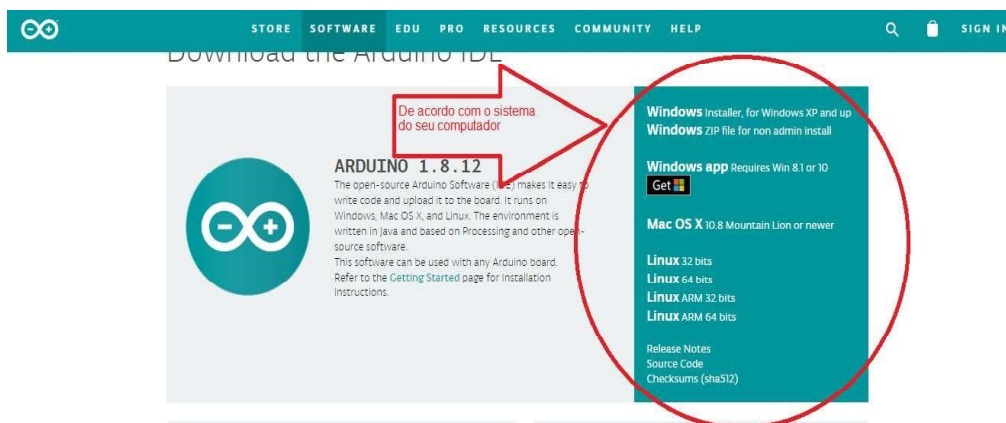
Fonte: Valmir Santos (2024)

- V. Para conectar, faça o download do aplicativo Arduino no endereço: <https://www.arduino.cc/>. Depois siga os passos abaixo:

1º passo: clique em Downloads.



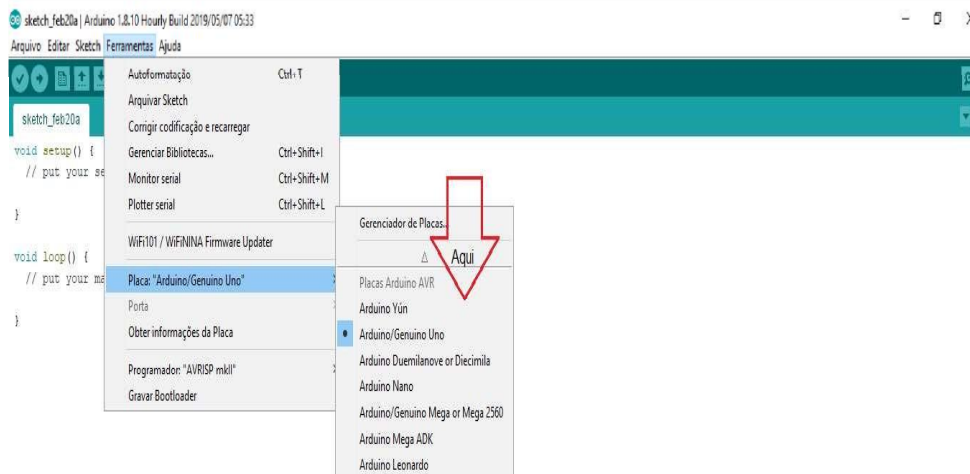
2º passo: escolha o sistema operacional.



3º passo: abra o programa na área de trabalho clicando duas vezes no ícone:



4º passo: seleccione a placa que será usada.



5º passo: copie o código abaixo.

```
#include <Servo.h> // The settings library can be found in the Arduino import library
```

```
int mode = 0; // Set the variable to mode
```

```
Servo servo_pin_6; // Create a servo object
```

```
Servo servo_pin_2; // Create a servo object
```

```
#include <LiquidCrystal.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
#include <Wire.h>
```

```
//Inicializa o display no endereço 0x27
```

```
LiquidCrystal_I2C lcd(0x27,16,2);
```

```
void setup() {
```



```
lcd.begin();

(16 ,2); //SETA A QUANTIDADE DE COLUNAS(16) E O NÚMERO DE LINHAS(2) DO
DISPLAY

lcd.setBacklight(HIGH); //LIGA O BACKLIGHT (LUZ DE FUNDO)

Serial.begin (9600); // Set the serial monitoring window data rate to 9600 bps

pinMode( 2 , INPUT); // Set the D pin input pin
servo_pin_6.attach(6); // Set the foot position to the desired position
servo_pin_2.attach(2); // Set the foot position to the desired position

}

void loop() {

Serial.print(analogRead( A5 )); // Print the reading pins of A5 on the monitoring window
Serial.print(" "); // space
Serial.println(); // Line breaks

if ( analogRead( A5 ) > 650 ) { // If the number of A5 pins is greater than 650
  mode = 0; // Set mode to 1

servo_pin_6.write( 0 ); // Perform to the right foot position
servo_pin_2.write( 0 ); // Perform to the right foot position
}

if ( analogRead( A5 ) < 650 && mode == 0 ) { // If the light sensitivity number (A5) is less
than 650 and mode 0
  mode = 1; // Set it to mode 1 when you come in

  if ( random( 0 , 2 ) == 0 ) { // Let the random sequence (0 or 1) be executed when it is
equal to 0
    servo_pin_6.write( 100 ); // The motor turns to angle 100
    delay( 500 ); // Stop for half a second, available or not
  }

  else { // If the above if is not reached, that is, the random sequence = 1 will be executed
this way
```



**GET UP
SCIENCE**



```
servo_pin_2.write( 100 ); // The motor turns to angle 100
delay( 500 ); // Stop for half a second, available or not
}
}
}
}
```

6º passo: cole o código na área de trabalho do programa, indicada a seguir.



```
sketch_feb20a | Arduino 1.8.10 Hourly Build 2019/05/07 05:33
Arquivo Editar Sketch Ferramentas Ajuda
sketch_feb20a
void setup() {
  // put your setup code here, to run once:
}
void loop() {
  // put your main code here, to run repeatedly:
}
```

7º passo: clique em verificar para saber se há algum erro.



```
sketch_feb20a | Arduino 1.8.10 Hourly Build 2019/05/07 05:33
Arquivo Editar Sketch Ferramentas Ajuda
sketch_feb20a
// projeto voltagem no LED para melhor consistência de float
int Vled = 20 ;
float escala = 100 * // 100 para volts, 0.1 para milivolts
void setup() {
  Serial.begin(9600);
  pinMode(Vled, OUTPUT);
}
void loop() {
  float tensao = analogRead(Vled);
  tensao = map(tensao, 0, 1023, 0, 500);
  float voltage = (tensao/escala);
  // Exibe o valor lido
  Serial.println(voltage); Serial.println(" V");
  delay(100);
}
```

8º passo: transfira o código após conectar o cabo USB (Universal Serial Bus ou “Porta Universal”).

```

sketch_feb15a | Arduino 1.8.10 Hourly Build 2019/05/07 05:33
Arquivo Editar Sketch Ferramentas Ajuda
sketch_feb15a.g
// projeto voltagem no I2C para medir constante de Planck

int vled = A0 ;
float escala = 100 // 100 para volts, 0.1 para milivolts

void setup() {
  Serial.begin(9400);
  pinMode(vled, INPUT);
}

void loop() {
  float tensao = analogRead(vled);
  tensao = map(tensao, 0, 1023, 0, 505);
  float voltage = (tensao/escala);
  // Exibe o valor lido
  Serial.println(voltage); Serial.println(" V");
  delay(100);
}

```

Upload complete

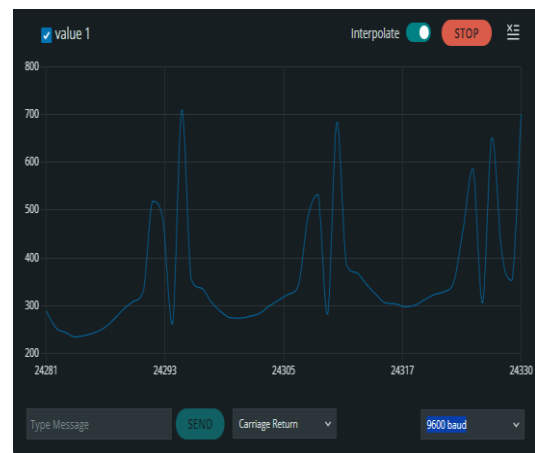
Sketch uses 3400 bytes (11%) of space for permanent programs. Maximum is 32254 bytes.
Variables global uses 330 bytes (1%) of memory. Maximum is 2048 bytes.
Variables local uses 194 bytes for variables local. Maximum is 2048 bytes.

9º passo: ligue o Arduino no computador com o código e o circuito montado na protoboard. Verifique a informação que será mostrada no Monitor serial. Através dessas imagens verifica-se que há passagem de tensão para o dispositivo.

```

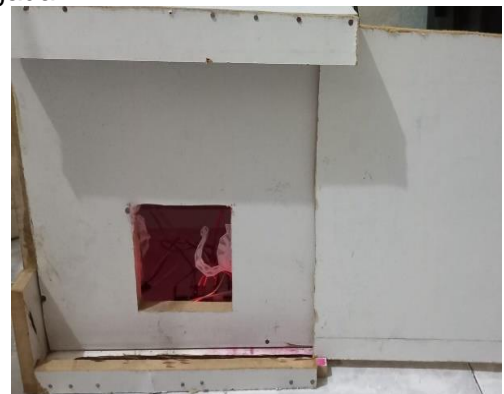
[message (Click to send message to Arduino Uno (1)) (Close)]
02:19:55.661 -> 487
02:19:55.661 -> 638
02:19:55.661 -> 564
02:19:55.661 -> 432
02:19:55.661 -> 735
02:19:56.108 -> 462
02:19:56.108 -> 447
02:19:56.108 -> 433
02:19:56.108 -> 425
02:19:56.108 -> 411
02:19:56.100 -> 417
02:19:56.155 -> 415
02:19:56.155 -> 420
02:19:56.155 -> 435
02:19:56.155 -> 444
02:19:56.155 -> 451
02:19:56.155 -> 466
02:19:56.155 -> 564
02:19:56.155 -> 610
02:19:56.155 -> 422
02:19:56.155 -> 720
02:19:56.652 -> 483
02:19:56.652 -> 465
02:19:56.652 -> 455

```



10º passo: Verifique o que ocorre com a representação do gato na caixa.

Caixa desligada e ligada



Fonte: Valmir Santos (2024).

Resultados e discussão

O experimento mental do Gato de Schrödinger exemplifica o conceito de superposição quântica, em que um sistema quântico, como o próprio gato, pode existir simultaneamente em múltiplos estados (vivo e morto), até que seja observado. Esse pensamento é fundamental para a compreensão da física quântica, onde partículas subatômicas podem estar em vários lugares ao mesmo tempo, interagindo de maneiras que desafiam a nossa lógica do cotidiano.

Se, ao invés de um gato, pensarmos em duas partículas subatômicas, ao serem observadas, elas deixam o estado de superposição e “decidem” em qual local estão, pois a observação colapsa esse estado. A física quântica sugere que o ato de observar interfere diretamente no comportamento das partículas, o que torna a realidade observável algo determinado pela própria ação de observação. No entanto, esses fenômenos ocorrem somente na escala microscópica, enquanto, no mundo macroscópico, as interações seguem as leis determinísticas da mecânica clássica, como observado no experimento da fenda dupla.

Neste contexto, o objetivo do experimento é explorar como a observação altera o estado quântico de uma partícula, ilustrando o princípio da incerteza, que impede a medição simultânea e precisa da posição e velocidade de uma partícula. A proposta fim do experimento é ajudar na compreensão dos fenômenos da física quântica, especialmente no que diz respeito à interferência e colapso do estado quântico quando uma medição é realizada. Isso visa fomentar o desenvolvimento de habilidades críticas no ensino de Física Moderna, integrando teoria e prática empírica.

Referências

METZGER, J. P.; RAABE, A. L.A.; SANTANA, A. L. M.; GOMES, E. B.; SOUZA, F. T. DE; RAMOS, G. L.; CUCCO, L. A.; VIEIRA, M. F. V. Características do Pensamento Computacional Desenvolvidas em Aprendizes do Ensino Médio por meio de Atividades Makers. In XXIII Workshop de Informática na Escola, p. 1-10, 2017. Disponível em: <https://sol.sbc.org.br/index.php/wie/article/view/16250>. Acesso em 25/12/2022.

SILVEIRA, S., GIRARDI, M. Desenvolvimento de um kit experimental com Arduino para o ensino de Física Moderna no Ensino Médio. Revista Brasileira de Ensino de



Física, vol. 39, nº 4, e4502 (2017). <https://doi.org/10.1590/1806-9126-RBEF-2016-0287>. 449.

SOUZA, A. R., PAIXÃO, A. C., UZEDA, D. D., DIAS, M. A., DUARTE, S. E AMORIM, H. S. (2011). A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC. *Revista Brasileira de Ensino de Física*, São Paulo, 33 (1), 1702-1-1702-5.