****

**Licenciatura em Engenharia Informática**

**Tecnologia e Arquitetura de Computadores 2023/2024**

**Trabalho Prático nº 2**

**Desenvolvimento de Aplicação para Controlo de Elevadores**

**Nelson Cunha a2023142681**

**Índice**

[**1. Introdução** 3](#_Toc161516762)

[**2. Métodos** 3](#_Toc161516763)

[**3. Capítulo 3: Resultados** 4](#_Toc161516764)

[**4. Discussão** 9](#_Toc161516765)

[**5. Conclusão** 9](#_Toc161516766)

[**6. Referências** 9](#_Toc161516767)

**1. Introdução**

Este trabalho tem como objetivo desenvolver uma aplicação para gestão de parque de estacionamento para a unidade curricular “Tecnologia e Arquitetura de Computadores ” , este trabalho foi executado com os conhecimentos adquiridos anteriormente (em todos os trabalhos do ano ) , e juntando todo esse conhecimento é possível.

**2. Métodos**

Para a realização do trabalho comecei por montar um circuito muito primitivo no tinkercard para começar a estruturar ideias na minha cabeça para daí prosseguir com a realização do algoritmo e fluxograma.

Depois da execução do algoritmo e do fluxograma finalizei a parte do tinkercard começando assim a parte do código. Como já tinha tudo estruturado e organizado foi mais fácil a realização do código para o mesmo porém, tive algumas dificuldades devido ao numero de variáveis e circuito mais complexo que requisitaram uma maior atenção e cuidado meu. No código foi usada um estratégia de código esquematizado dividido varias funções .

Para a realização do tinker card foi utilizado os seguintes componentes:

* 1 breadboard
* 1 piezos (alarmes sonoros)
* 1 leds amarelo
* 2 micro servos
* 2 Arduíno Uno R3
* 10 resistências 1k Ω
* 1 visor de 7 segmentos
* 2 butões
* Diversos cabos para efetuar as ligações

Os testes que fiz para validar o projeto foi simular a subida e descida do elevador

**Materiais utilizados :**

Uma imagem com texto, captura de ecrã, software, número

Descrição gerada automaticamente

**Desenho do circuito :**

**Arduino 1 :**

**Uma imagem com texto, diagrama, Paralelo, Esquema

Descrição gerada automaticamente**

**Arduino2 Uma imagem com file, texto, diagrama, Gráfico

Descrição gerada automaticamente**

**3. Capítulo 3: Resultados**

Com a junção dos materiais citados anteriormente segue-se a seguinte montagem no tinkercard:

Uma imagem com texto, Engenharia eletrónica, eletrónica, Componente de circuito

Descrição gerada automaticamente

Como dá para observar esta é a minha montagem do tinkercard acerca o projeto.

Podemos ver que na parte superior existe dois micro servos que servem como fossem as portas do elevador debaixo dos microservos temos um piezo ( alarme sonoro ) que serve para fazer barulho quando tu tentas ir para o mesmo andar que já te encontras debaixo do alarme tens um led amarelo que liga quando o elevador está em movimento , á direita encontra-se dois butões que representa o andar 1 e o andar 2 mais á direita na outra breadboard verificamos a existência de um visor de 7 segmentos que tem como objetivo mostrar o andar em que o elevador se encontra .

Uma imagem com texto, Engenharia eletrónica, eletrónica, diagrama

Descrição gerada automaticamente

Neste teste pusemos o elevador a subir para o 2 andar porem como as pessoas precisam de entrar as portas tem que abrir para as pessoas entrarem daí o microservo estar a 90º



Uma imagem com Engenharia eletrónica, circuito, Componente de circuito, Placa de teste

Descrição gerada automaticamente

Neste caso verificamos que a led está acessa ou seja está o elevador está em movimento



Uma imagem com Engenharia eletrónica, circuito, eletrónica

Descrição gerada automaticamente

Neste teste o elevador já se encontra no 2 andar .

**Algoritmo do código :**

No inicio crio variáveis e dou valores ás mesmas , sejam elas para atribuir pinos ou até para usar com uma bool.

Na função setup defino os pinos se é saída ou entrada

Começo o serial.

Depois do setup crio varias funções que servem para meter o mecanismo a funcionar , seja abrir porta, fechar porta , exibir andar mover para andar etc.

Tenho também comunicação entre dois arduinos utilizando o protocolo I2C

Depois da função loop “chamo” pelas funções para obter o resultado desejado.

**Segue-se então o código arduino 1**

#include <Servo.h>

#include <Wire.h>

const int butao\_elevador\_1 = 2;

const int butao\_elevador\_2 = 3;

const int led\_elevador = 4;

const int alarme = 5;

const int porta\_elevador = 6;

const int porta\_elevador2 = A0;

const int segA = 7;

const int segB = 8;

const int segC = 9;

const int segD = 10;

const int segE = 11;

const int segF = 12;

const int segG = 13;

volatile bool chamarAndar1 = false;

volatile bool chamarAndar2 = false;

int andarAtual = 1;

int viagens = 0;

Servo portaServo;

Servo porta2Servo;

void setup() {

pinMode(butao\_elevador\_1, INPUT\_PULLUP);

pinMode(butao\_elevador\_2, INPUT\_PULLUP);

pinMode(led\_elevador, OUTPUT);

pinMode(alarme, OUTPUT);

portaServo.attach(porta\_elevador);

portaServo.write(0);

porta2Servo.attach(porta\_elevador2);

porta2Servo.write(0);

pinMode(segA, OUTPUT);

pinMode(segB, OUTPUT);

pinMode(segC, OUTPUT);

pinMode(segD, OUTPUT);

pinMode(segE, OUTPUT);

pinMode(segF, OUTPUT);

pinMode(segG, OUTPUT);

Serial.begin(9600);

Serial.println("Sistema de controle do elevador iniciado.");

Wire.begin();

attachInterrupt(digitalPinToInterrupt(butao\_elevador\_1), chamarAndar1Handler, FALLING);

attachInterrupt(digitalPinToInterrupt(butao\_elevador\_2), chamarAndar2Handler, FALLING);

exibirAndar();

}

void loop() {

if (chamarAndar1) {

chamarAndar1 = false;

moverParaAndar(1);

}

if (chamarAndar2) {

chamarAndar2 = false;

moverParaAndar(2);

}

}

void chamarAndar1Handler() {

chamarAndar1 = true;

}

void chamarAndar2Handler() {

chamarAndar2 = true;

}

void moverParaAndar(int andar) {

if (andarAtual == andar) {

digitalWrite(alarme, HIGH);

abrirPorta();

delay(1000);

digitalWrite(alarme, LOW);

delay(4000);

fecharPorta();

Serial.print("Elevador parou no andar ");

Serial.println(andarAtual);

} else {

abrirPorta();

delay(5000);

fecharPorta();

Serial.print("Elevador iniciando deslocamento para o andar ");

Serial.println(andar);

digitalWrite(led\_elevador, HIGH);

Wire.beginTransmission(25);

Wire.write('x');

Wire.endTransmission();

delay(5000);

digitalWrite(led\_elevador, LOW);

andarAtual = andar;

abrirPorta();

Serial.print("Elevador parou no andar ");

Serial.println(andarAtual);

delay(5000);

fecharPorta();

}

exibirAndar();

}

void abrirPorta() {

if (andarAtual == 1) {

for (int pos = 0; pos <= 90; pos += 1) {

portaServo.write(pos);

delay(15);

}

} else if (andarAtual == 2) {

for (int pos = 0; pos <= 90; pos += 1) {

porta2Servo.write(pos);

delay(15);

}

}

}

void fecharPorta() {

if (andarAtual == 1) {

for (int pos = 90; pos >= 0; pos -= 1) {

portaServo.write(pos);

delay(15);

}

} else if (andarAtual == 2) {

for (int pos = 90; pos >= 0; pos -= 1) {

porta2Servo.write(pos);

delay(15);

}

}

}

void exibirAndar() {

if (andarAtual == 1) {

digitalWrite(segA, LOW);

digitalWrite(segB, HIGH);

digitalWrite(segC, HIGH);

digitalWrite(segD, LOW);

digitalWrite(segE, LOW);

digitalWrite(segF, LOW);

digitalWrite(segG, LOW);

} else if (andarAtual == 2) {

digitalWrite(segA, HIGH);

digitalWrite(segB, HIGH);

digitalWrite(segC, LOW);

digitalWrite(segD, HIGH);

digitalWrite(segE, HIGH);

digitalWrite(segF, LOW);

digitalWrite(segG, HIGH);

}

}

**Segue-se então o código arduino 2**

#include <Wire.h>

unsigned long total\_viagens = 0;

unsigned int viagens\_ultimo\_minuto = 0;

unsigned long iniciotimer = 0;

unsigned long timer\_antigo = 0;

const long intervalo = 60000;

unsigned long minutos\_total = 0;

void setup() {

Wire.begin(25);

Wire.onReceive(receberEvento);

Serial.begin(9600);

iniciotimer = millis();

}

void loop() {

unsigned long correnteMillis = millis();

if (correnteMillis - timer\_antigo >= intervalo) {

timer\_antigo = correnteMillis;

minutos\_total++;

enviarMetricasParaComputador();

viagens\_ultimo\_minuto = 0;

}

}

void receberEvento(int quantos) {

while (Wire.available()) {

char c = Wire.read();

if (c == 'x') {

total\_viagens++;

viagens\_ultimo\_minuto++;

}

}

}

void enviarMetricasParaComputador() {

float media\_viagens\_minuto;

if (minutos\_total > 0) {

media\_viagens\_minuto = (float)total\_viagens / minutos\_total;

} else {

media\_viagens\_minuto = 0;

}

Serial.print("Total de viagens: ");

Serial.println(total\_viagens);

Serial.print("Viagens no último minuto: ");

Serial.println(viagens\_ultimo\_minuto);

Serial.print("Média de viagens por minuto: ");

Serial.println(media\_viagens\_minuto);

}

**4. Discussão**

Agora que finalizei o trabalho posso dizer que a realização do fluxograma ajuda e muito na realização do código, porém mesmo assim enfrento um problema no meu código que sei porque é que acontece, mas não sei resolver.

Apesar disso acho que o trabalho foi bem executado e de forma eficiente mas com aquele problema que pode levar a erros pontuais, mas que nada seja impossível de resolver.

Para um futuro talvez pensava em organizar melhor o tinkercard caso queira aumentar o numero de slots de andares .

**5. Conclusão**

Em suma, apesar dos desafios enfrentados, considero que o trabalho foi bem executado.

Senti que adquiri as capacidades suficientes para a realização deste projeto mas sinto que necessito de mais prática.

Tempo gasto por semana ula – 3 horas semanais de aula prática +-

Tempo gasto por semana extra-aula – entre 3-5 horas

[Link TinkerCard](https://www.tinkercad.com/things/lAQ3sJUg3J0-fabulous-snaget-jaagub/editel?sharecode=l5pK_IIp4eta6nBsRfBhwBb_ECQ4kHE9xva0y3rG96I)

(

[link](https://drive.google.com/file/d/1JY2m6Sn2IWhdEuUVIz4Ce8XxmdTov_R9/view?usp=sharing) [fluxo](https://drive.google.com/file/d/1cxcFYGLblWZNWCk3t0xgJ5GeVdHOQC9p/view?usp=sharing)

**6. Referências**

Para me ajudar na realização deste projeto usei o os trabalhos anteriormente utilizados e os documentos de PDF que continham alguma informação sobre projetos anteriores.

Usei o conhecimento anteriormente obtido das unidades curriculares Fundamentos de Programação e Sistemas Digitais.

E procurei informação no tinkercard fazendo um tutorial que os mesmos disponibilizam acerca o wire.