

**Relatório**

**Sistemas Embebidos em Tempo Real**

**Alunos:**

**João Fernandes Nº18825**

**António Oliveira Nº18833**

**Carlos Martins Nº18836**

**Professor: Paulo Macedo**

**Licenciatura em Engenharia de Sistema Informáticos**

Barcelos, janeiro, 2021

Índice

[Índice de Figuras 3](#_Toc93078828)

[Introdução 3](#_Toc93078829)

[Análise de Requisitos 4](#_Toc93078830)

[Especificação do sistema 8](#_Toc93078831)

[Desenvolvimento da Arquitetura 8](#_Toc93078832)

[Arquitetura Sistema A 9](#_Toc93078833)

[Arquitetura Sistema B 9](#_Toc93078834)

[Arquitetura Sistema C 10](#_Toc93078835)

[Arquitetura Sistema D 10](#_Toc93078836)

[Modelo de Conceção 11](#_Toc93078837)

[Construção do sistema 12](#_Toc93078838)

[Sistema A – Controlo de iluminação interior 12](#_Toc93078839)

[Código 13](#_Toc93078840)

[Sistema B – Controlo de climatização 14](#_Toc93078841)

[Código 15](#_Toc93078842)

[Sistema C – Sistema de acesso ao estacionamento (Físico) 17](#_Toc93078843)

[Código 18](#_Toc93078844)

[Sistema C – Sistema de acesso ao estacionamento (Tinkercad) 20](#_Toc93078845)

[Código 21](#_Toc93078846)

[Sistema D – Sistema de segurança (Tinkercad) 22](#_Toc93078847)

[Código 22](#_Toc93078848)

[Sistema D – Sistema de segurança (Físico) 23](#_Toc93078849)

[Testes/Resultados 24](#_Toc93078850)

[Conclusão 25](#_Toc93078851)

# Índice de Figuras

[Figura 1 - Arquitetura Sistema A 9](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078852)

[Figura 2 - Arquitetura Sistema B 9](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078853)

[Figura 3 - Arquitetura Sistema C 10](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078854)

[Figura 4 - Arquitetura Sistema D 10](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078855)

[Figura 5 - Modelo Waterfall 11](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078856)

[Figura 6 - Construção do Sistema A 12](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078857)

[Figura 7 - Construção do Sistema C 17](file:///C:\Users\João%20Fernandes\Desktop\TP_SETR_18825_18833_18836.docx#_Toc93078858)

[Figura 8 - Construção do sistema C (Tinkercad) 20](#_Toc93078859)

[Figura 9 - Construção do Sistema D (Tinkercad) 22](#_Toc93078860)

[Figura 10 - Construção do sistema D (Físico) 23](#_Toc93078861)

# Introdução

Este trabalho tem como objetivo o desenvolvimento de um conjunto de sistemas embebidos em tempo Real integrados para uma Smart Home, ou seja, um projeto piloto Home Automation composto por vários sistemas embebidos, cada um com requisitos e funcionalidades específicas (iluminação, climatização, parking e segurança). Para desenvolver o trabalho utilizamos um IDE de Arduíno para fazer a programação na linguagem C. Para alguns exercícios usamos o “TinkerCad” como o professor pediu. Tendo em conta isto, este relatório será realizado à medida que a resolução do trabalho é feita, de forma a organizar melhor os conteúdos.

# Análise de Requisitos

Projeto “Home Automation” para realizar funcionalidades específicas: Iluminação, Climatização, Parking e Segurança.

|  |  |
| --- | --- |
| Sistema | Requisitos Funcionais |
| A | No controle a luminosidade do espaço interior, em função da luz solar, é regulado através de um sensor, a luminosidade, garantindo uma iluminação constante e uma maior eficiência energética. |
| B | Para realizar o controlo da climatização, uma ventoinha é acionada para o arrefecimento do espaço em função da temperatura fornecidos pelo sensor de temperatura. |
| C | Para o sistema de parking, um controlo remoto controla a barra que abre e fecha, o acesso ao parque de estacionamento. |
| D | Para o sistema de segurança, um sensor de movimento deteta o movimento de intrusos, acionando um sinal luminoso e sonoro. |

|  |  |
| --- | --- |
| Sistema | Requisitos Não-Funcionais |
| A | Inputs: Sensor LDR.  Outputs: Leds de iluminação interior.  Funcionalidade: Minimiza custos de eletricidade.  Interface com o utilizador: Sensor LDR que regula a intensidade da luz e um LED verde que indica uma temperatura regulada e um LED vermelho que indica que a ventoinha se encontra em funcionamento.  Performance: Otimização e redução de custo da eletricidade.  Propósito: Transmitir conforto e otimização dos custos de energia. |
| B | Inputs: LED vermelho e verde, sensor temperatura.  Outputs: LCD 16 X 2 que mostra a temperatura e o mostra o estado da ventoinha.  Funcionalidade: Climatização dentro de casa.  Interface com o utilizador: O controlo de climatização através de uma ventoinha, e um LCD que mostra a temperatura e o estado da ventoinha.  Performance: Atualização do ecrã LCD para saber a temperatura ambiente.  Propósito: Manter a temperatura ambiente. |
| C | Inputs: Controle Remoto infravermelhos.  Outputs:  Funcionalidade: Sistema útil para estacionamento de forma a controlar remotamente.  Interface com o utilizador: Para o acesso ao estacionamento, existe um controlo remoto, onde o utilizador consegue controlar a barra de acesso.  Performance: Sensor que deteta movimento a uma certa distância evitando acidentes graves.  Propósito: Facilidade no estacionamento. |
| D | Inputs: Botão de desarme do alarme.  Outputs: Sinal luminoso (led) e sinal sonoro característico de um alarme.  Funcionalidade: Melhor segurança na casa.  Interface com o utilizador: Para o sistema de segurança o utilizador consegue pressionar num botão para desarmar o alarme.  Performance: Sensibilidade do sensor para detetar movimentos mais precisos.  Propósito: Aumentar a segurança da casa |

# Especificação do sistema

**Deve incluir:**

* A informação sobre a luminosidade e a temperatura ambiente no interior da casa;
* Movimentos bruscos quando ligado o sistema de segurança;
* Dados sobre a temperatura após atuar o sistema de climatização;
* LCD 16 X 2 que mostra os dados, botão de desarme e controlo remoto;
* Realizar a climatização e o controlo de luminosidade, seja de noite ou de dia, transmitir maior segurança quando o utilizador permanece ausente e a utilidade de uma barra de acesso para o estacionamento do automóvel;
* Sensores de temperatura, movimento e de luminosidade são sistemas que ficam em execução para que os sistemas funcionem.

# Desenvolvimento da Arquitetura

* Todos os componentes utilizados servirão para realizar uma “home automation”, tornando uma casa comum, numa casa inteligente. Se todos os requisitos e especificações satisfazerem as condições, teremos os sistemas pretendidos.
* Arduíno, sensor de temperatura, movimento e luminosidade, cabos, motor servo, ventoinha, display LCD 16 X 2, botão (pressão), sinal sonoro e luminoso e por fim leds.
* Arduíno IDE e TinkerCad

## Arquitetura Sistema A

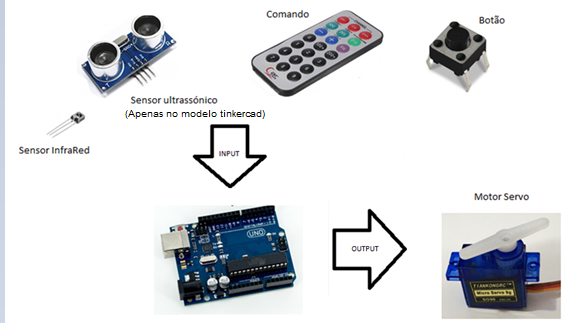
Figura 1 - Arquitetura Sistema A

## Arquitetura Sistema B

Figura 2 - Arquitetura Sistema B

## Arquitetura Sistema C

Figura 3 - Arquitetura Sistema C



## Arquitetura Sistema D

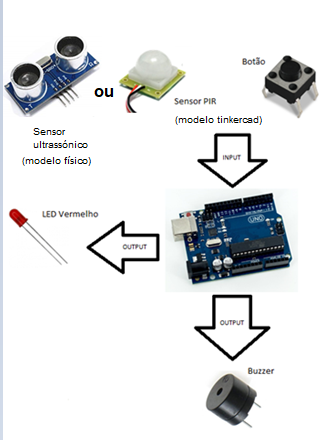


Figura 4 - Arquitetura Sistema D

# Modelo de Conceção

O modelo utilizado no trabalho prático foi o ***waterfall****.* Foram definidos para o desenvolvimento dos Sistemas uma abordagem ***Top-down***:

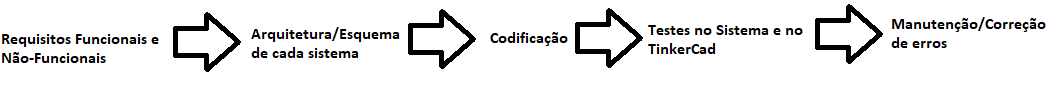
* **Requisitos** funcionais e não funcionais para cada sistema;
* Uma **arquitetura** com um esquema prévio para o desenvolvimento de cada sistema;
* A implementação de um **código** que conseguisse cumprir o que foi imposto inicialmente;
* **Testes**, tanto do código, como na montagem do circuito como no simulador TinkerCad;
*  Por fim uma **manutenção** para a correção de alguns erros, otimização do sistema e atualização do código.

Figura 5 - Modelo Waterfall

# Construção do sistema

## Sistema A – Controlo de iluminação interior

Para simular este sistema, utilizamos um LED e um sensor LDR. O objetivo é controlar a luminosidade do LED de acordo com a iluminação do espaço. Para saber qual a luminosidade do LED, foi definido escalas de intensidade de luz de forma a que, conforme a iluminação do espaço, detetado pelo sensor LDR, o LED ligue e permaneça com uma certa intensidade de luz. Os valores atribuídos para o sensor LDR foram:

* para o LED não ligar - < 200;
* para uma intensidade de 64 - ≥ 200 e < 500;
* para uma intensidade de 128 - ≥ 500 e < 800;
* para uma intensidade de 255 - ≥ 800;

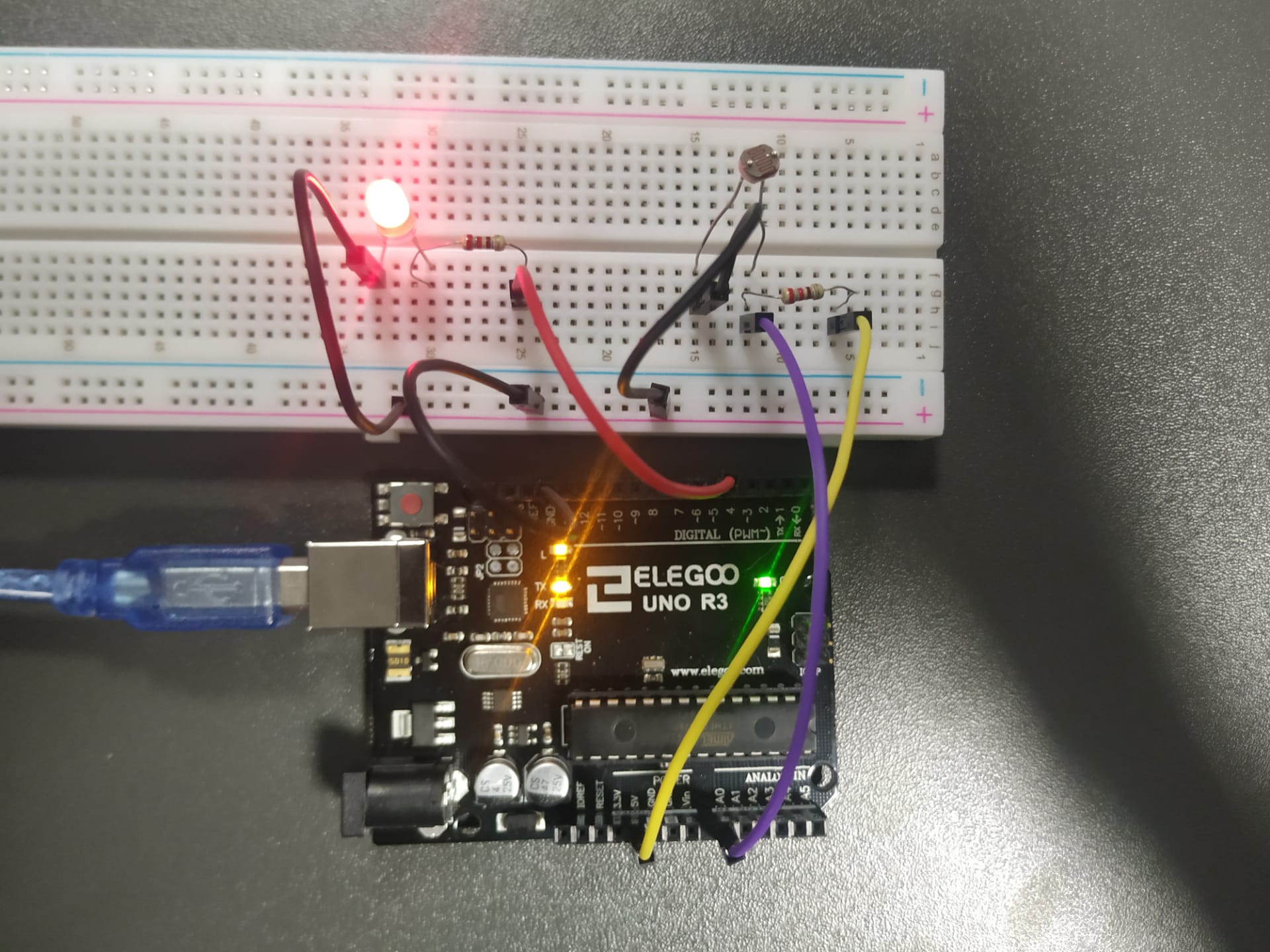


Figura - Construção do Sistema A

### Código



## Sistema B – Controlo de climatização

Neste sistema é pretendido desenvolver um controlo de temperatura ambiente através de uma ventoinha que é acionada para arrefecer o espaço onde se situa, em função dos valores de temperatura que são obtidos pelo sensor de temperatura.

Quando o sensor de temperatura detetar 25 graus celsius a ventoinha liga e desliga sempre que a temperatura é inferior a 20 graus celsius. Para saber quando está a arrefecer o LED vermelho liga e quando a temperatura estabiliza o LED verde liga. Na 1ª linha do LCD mostra o estado da ventoinha, se ela está ON ou OFF e na 2ª linha mostra a temperatura atual. O potenciômetro regula a luminosidade do Display LCD.

Desenvolvemos este exercício no tinkercad porque o LCD e o sensor de temperatura que tínhamos disponíveis não estavam a funcionar corretamente.

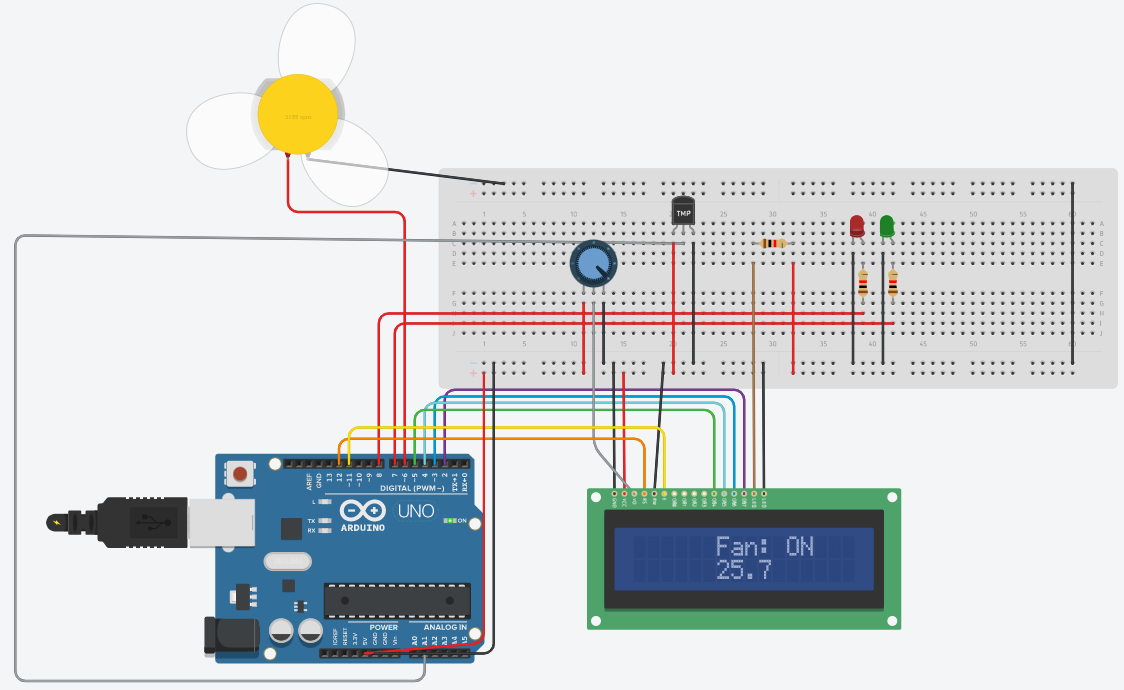


Figura 7 - Construção do sistema B

### Código





## Sistema C – Sistema de acesso ao estacionamento (Físico)

Este sistema permite que um comando controle uma barra de acesso a um parque de estacionamento. Na simulação, o comando infrared controla o motor servo da seguinte forma:

* O botão 0 do comando levanta a barra 90º verticalmente;
* O botão 1 do comando desce a barra até aos 0º horizontalmente;
* O botão 2 do comando suspende a barra.

De modo a reduzir a velocidade do motor Servo (que simula a barra de entrada/saída de um estacionamento) utilizamos um ciclo for em que a posição do mesmo altera a cada 25ms.

Uma imagem com texto

Descrição gerada automaticamente

Figura - Construção do Sistema C

### Código







## Sistema C – Sistema de acesso ao estacionamento (Tinkercad)

Desenvolvemos também este sistema em Tinkercad de modo a implementarmos um sensor ultrassónico na sua construção. Este sensor ultrassónico verifica se não está ninguém dentro de uma distância definida por nós (2 metros) e se a condição for satisfeita o servo começa a descer, se não o servo não desce, além disso, se a condição for satisfeita, mas de repente entrar alguém dentro do radar de deteção o servo começa automaticamente a subir.

A justificação para implementar o sensor ultrassónico neste sistema no Tinkercad e não fisicamente foi que o sensor ultrassónico estava a causar algum tipo de interferência com o sensor IR. Supomos que a causa disto seja devido ao número de cálculos que o sistema tem de executar para calcular a distância a que um objeto se encontra do sensor ultrassónico a cada vez que o servo está a descer.

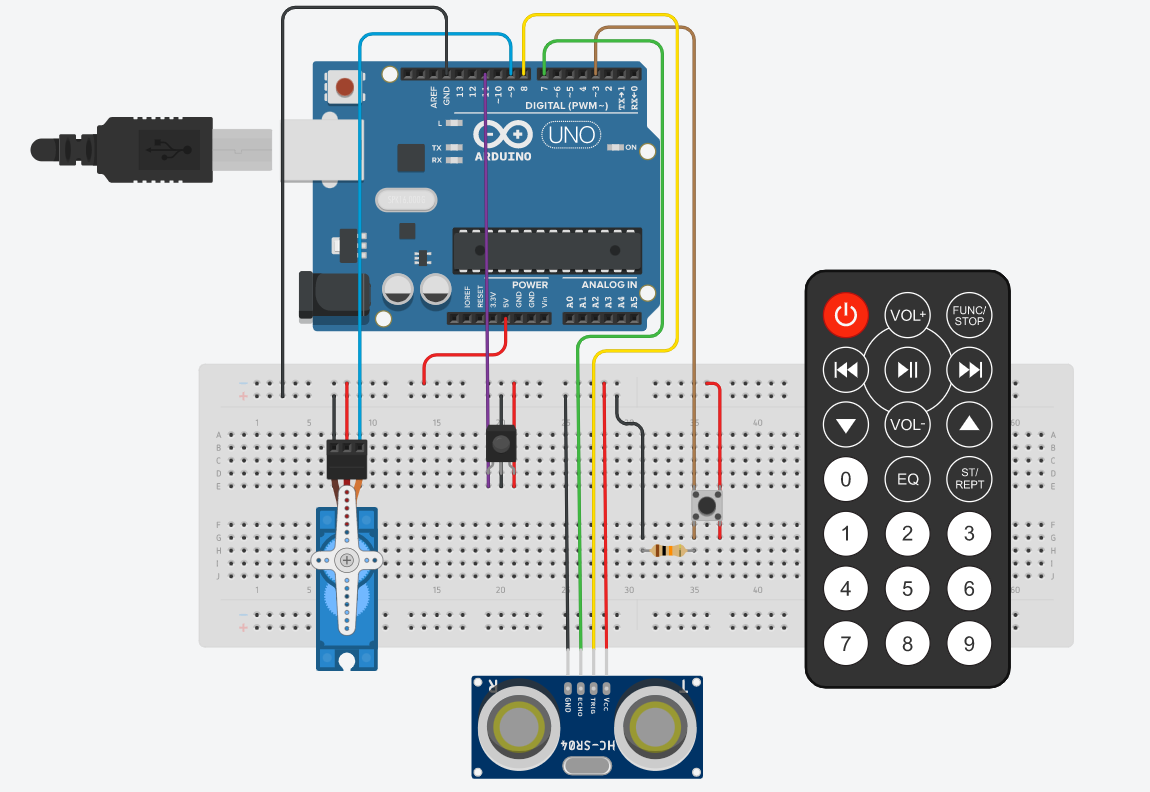


Figura - Construção do sistema C (Tinkercad)

### Código

Para manter não estar a mostrar praticamente o mesmo código vamos só deixar apenas as alterações feitas o código anterior.



## Sistema D – Sistema de segurança (Tinkercad)

Foi criado um sistema de segurança para a deteção de movimentos através de um sensor PIR. No momento que deteta movimento é acionado um sinal luminoso através de um LED vermelho, um sinal sonoro que seja característico de um alarme e dura 10 segundos e um botão que permita desarmar o alarme. Vale a pena notar que este sinal sonoro pode parecer um pouco distorcido no tinkercad, porque no tinkercad um segundo não equivale realmente a um segundo verdadeiro.

Como o arduino não suporta multiprocessamento tivemos que utilizar multitasking (simulamos isto com a função millis) para conseguirmos obter o LED a piscar e um sinal sonoro em simultâneo.

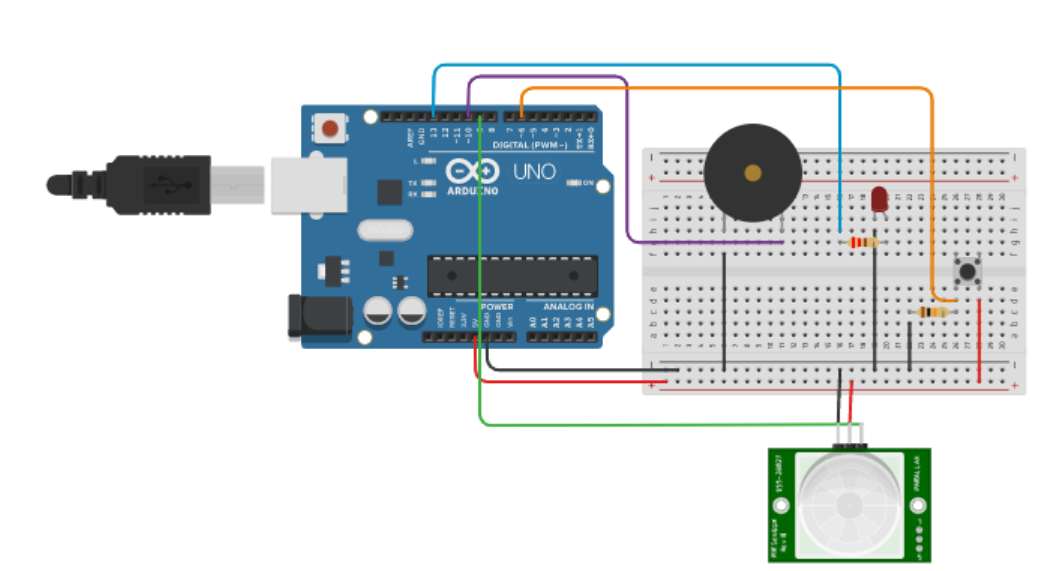


Figura - Construção do Sistema D (Tinkercad)

### Código

## Sistema D – Sistema de segurança (Físico)

Fisicamente, temos praticamente o mesmo sistema sendo que a única diferença é que tivemos de substituir o sensor IR por um sensor ultrassónico, porque não tínhamos nenhum disponível a funcionar. Isto causou ligeiras diferenças no código porque o sensor é acionado a partir de uma certa distância e não quando deteta um movimento em todo o seu radar de deteção.

Uma imagem com eletrónica

Descrição gerada automaticamente

Figura - Construção do sistema D (Físico)

# Testes/Resultados

Para o sistema A, como montámos fisicamente, gravámos um vídeo em que é possível verificar a composição do circuito assim como os testes realizados ao mesmo. Os testes realizados foram apontar a lanterna de um telemóvel ao sensor LDR a diferentes distâncias para a intensidade detetada ser maior ou menor dependendo da distância. [Vídeo](https://youtube.com/watch?v=xpFDTRPN_Lw&feature=share)

Para o sistema B não o montámos fisicamente pelas razões já abordadas nos tópicos anteriores, mas como desenvolvemos o circuito no tinkercad o mesmo está acessível e testável a partir do seguinte link: [tinkercad](https://www.tinkercad.com/things/loYkcqU1ubO-copy-of-sistema-b/editel?sharecode=znh1QBN4QtB7fGRnaE_AeQvW8941W1s1lN-vgcVpSRA)

Para o sistema C foi realizada uma versão no tinkercad ([tinkercad](https://www.tinkercad.com/things/didBD26oC0C-ex3-normal-tinkercad/editel?sharecode=_XAf9850TJ4BJ5_e_uUqKAF70wARMVVNjwWy-bNJuqA)) e uma versão física ([vídeo](https://www.youtube.com/watch?v=eIupzTW1fqo)). Os testes realizados para o modelo físico foram subir o servo, descer o servo e pressionar o botão para interromper o movimento do servo. No tinkercad temos praticamente o mesmo circuito, mas com ligeiras diferenças no código porque incorporámos um sensor ultrassónico que deteta se existe algo dentro do campo de deteção definido que neste caso são 2 metros.

Para o sistema D também desenvolvemos uma versão física ([vídeo](https://www.youtube.com/watch?v=tKYVAf5rVbI)) e uma versão no tinkercad ([tinkercad](https://www.tinkercad.com/things/exO1Rk5SdXS-surprising-uusam-amberis/editel?sharecode=rTVYjeHTbFulCLccr0waXMHXzPds88VgXcFlXotC80E)). A diferença da versão física para a do tinkercad é que na versão física substituímos o sensor PIR por um sensor ultrassónico (e algum código por causa da diferença de sensores). Os testes realizados para a versão física foram passar a mão a uma distância detetável pelo sensor ultrassónico e desativar o alarme no botão.

# Análise de performance do sistema C

De forma a medir a performance de uma forma constante sem variáveis que podem interferir na medição, foi feita a medição de performance numa função que não tem interferência de um utilizador e não depende de funções externas.

Na função de descer a barra, “descerBarra()”, criamos uma variável que irá guardar o tempo desde a execução do programa dado pela função “micros()” e após a execução de todo o código da função, guardamos numa nova variável o tempo no fim da execução de forma a calcular o tempo que demorou a executar, utilizando o cálculo tempo final – tempo inicial.

Como podemos ver no código A o tempo de execução foi 3420176 microssegundos, ou seja, maior devido a fatores como a instanciação de uma função auxiliar, ao contrário do código B que mantém o código sem utilizar uma função auxiliar, obtendo um tempo de execução de 3411556 microssegundos.

Código A



Código B





# Conclusão

Chegamos ao fim do semestre e cremos que foi uma disciplina trabalhosa, mas que trabalhamos com todo o gosto. Foi uma disciplina diferente das outras todas que estávamos habituados, porque nunca antes tínhamos mexido em circuitos eletrónicos com a liberdade que nos foi dada nesta cadeira e ainda exploramos e conhecemos componentes que não conhecíamos.

Foi também um trabalho em que deu para usar no nosso "instinto de engenheiro", porque passamos algumas dificuldades com a falta de componentes funcionais, mas demos a volta a isso utilizando outros componentes de modo a cumprir o mesmo objetivo, como por exemplo, no sistema D a utilização de um sensor ultrassónico no lugar de um sensor PIR.

Para finalizar concluímos que o trabalho foi uma mais-valia muito grande porque conseguimos desenvolver o nosso conhecimento e aplicá-lo praticamente todo, senão todo o conteúdo lecionado nas aulas.