

Claudio Vega Dourado Filho

# **Sistema de Automação Predial**

Campina Grande, Brasil

1 de novembro de 2016

Claudio Vega Dourado Filho

## **Sistema de Automação Predial**

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Campus Campina Grande, como parte dos requisitos necessários para obtenção do título de Graduado em Engenharia Elétrica.

Univesidade Federal de Campina Grande - UFCG  
Centro de Engenharia Elétrica e Informática - CEEI  
Departamento de Engenharia Elétrica - DEE

Orientador: George Acioli Junior

Campina Grande, Brasil

1 de novembro de 2016

---

Claudio Vega Dourado Filho

Sistema de Automação Predial/ Claudio Vega Dourado Filho. – Campina Grande, Brasil, 1 de novembro de 2016-  
26 p. : il. ; 30 cm.

Orientador: George Acioli Junior

Trabalho de Conclusão de Curso – Univesidade Federal de Campina Grande - UFCG  
Centro de Engenharia Elétrica e Informática - CEEI  
Departamento de Engenharia Elétrica - DEE , 1 de novembro de 2016.

---

Claudio Vega Dourado Filho

## **Sistema de Automação Predial**

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Campus Campina Grande, como parte dos requisitos necessários para obtenção do título de Graduado em Engenharia Elétrica.

Trabalho aprovado. Campina Grande, Brasil, 1 de novembro de 2016:

---

**George Acioli Junior**  
Orientador

---

**Péricles Rezende Barros**  
Convidado

Campina Grande, Brasil  
1 de novembro de 2016

*Este trabalho é dedicado à minha família.*

# Agradecimentos

Agradeço a meus pais Claudio e Jucileide pelo apoio incondicional e motivação em todos os momentos do curso e da minha vida.

Aos amigos feitos no curso, que tornaram difíceis temporadas de estudo extremamente fáceis e até gratificante.

A minha namorada Nicole pelo companheirismo no estudo, amor e paciência.

Aos meus professores os quais contribuíram para minha formação como aluno. Em especial aos professores Péricles Rezende e George Aciole pela oportunidade de trabalhar no LIEC.

*"O único lugar aonde o sucesso vem antes do trabalho é no dicionário"*  
*(Albert Einstein)*

# Resumo

Este projeto é o resultado de um Trabalho de Conclusão de Curso realizado junto ao Departamento de Engenharia Elétrica da Universidade Federal de Campina Grande e ao Laboratório de Instrumentação Eletrônica e Controle (LIEC). Consiste na criação de um sistema de automação predial capaz de controlar a temperatura, umidade e luminosidade de uma sala.

**Palavras-chaves:**automação predial;eficiência energética;home automation.



# Abstract

This project is a result of a Senior Design Project held at the Department of Electrical Engineering at the Federal University of Campina Grande and at the Electronic Instrumentation and Control Laboratory (LIEC). It is the study and development of creation of a Building Automation System Able to control the temperature, humidity and brightness of a room.

**Key-words:**building automation;energy efficiency;Key-words

# Lista de ilustrações

Figura 1 – Esquema do sistema . . . . .	3
Figura 2 – Arduino/Genuino MKR1000 . . . . .	5
Figura 3 – Esquema de conexões do RHT03 . . . . .	6
Figura 4 – Pinagem do RHT03 . . . . .	6
Figura 5 – LDR . . . . .	7
Figura 6 – Esquema de ligação do LDR ao microcontrolador . . . . .	7
Figura 7 – Módulo Cartão SD Card . . . . .	8
Figura 8 – Módulo Relé . . . . .	8
Figura 9 – Módulo ESP8266 ESP-12F . . . . .	9
Figura 10 – Módulo ESP8266 ESP-01 . . . . .	9
Figura 11 – Adaptador USB serial TTL . . . . .	10
Figura 12 – Esquema de uma conexão MQTT . . . . .	11
Figura 13 – Fluxograma da lógica do módulo 1 . . . . .	13
Figura 14 – Fluxograma da lógica do módulo 2 . . . . .	14
Figura 15 – Ligação three way . . . . .	15
Figura 16 – Código infra vermelho . . . . .	15
Figura 17 – Fluxograma da lógica do módulo 3 . . . . .	16
Figura 18 – Aba Conexão com módulo . . . . .	17
Figura 19 – Aba Comandos . . . . .	18
Figura 20 – Aba Preferências . . . . .	19
Figura 21 – Aba Comunicação . . . . .	19
Figura 22 – Rede MQTT . . . . .	20
Figura 23 – Esquema do módulo 1 . . . . .	21
Figura 24 – Módulo 1 . . . . .	22
Figura 25 – Esquema do módulo 2 . . . . .	22
Figura 26 – Módulo 2 . . . . .	23
Figura 27 – Esquema do módulo 3 . . . . .	23
Figura 28 – Regulador 3.3V . . . . .	24
Figura 29 – Módulo 3 . . . . .	24

# Lista de tabelas

Tabela 1 – Característica do MKR1000 . . . . .	5
--	---

# Lista de abreviaturas e siglas

MQTT	Message Queue Telemetry Transport
LDR	Light Dependent Resistor
IoT	Internet of Things
LDR	Light Dependent Resistor
PCI	Placa de Circuito Impresso
IDE	Integrated development Environment
UFCG	Universidade Federal de Campina Grande
LIEC	Laboratório de Instrumentação Eletrônica e Controle

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	Objetivos	1
1.2	Motivação	1
1.3	Metodologia	1
1.4	Estrutura do documento	2
<b>2</b>	<b>SISTEMA</b>	<b>3</b>
2.1	Especificação do Módulos	3
2.1.1	Módulo 1	3
2.1.1.1	Arduino/Genuino MKR1000	4
2.1.1.2	Sensor de Temperatura e Umidade RHT03	5
2.1.1.3	LDR	6
2.1.1.4	Módulo Cartão SD Card	7
2.1.1.5	Módulo relé	8
2.1.2	Módulos 2 e 3	9
2.1.3	Adaptador USB serial TTL	9
2.1.4	Protocolo MQTT	10
<b>3</b>	<b>DESENVOLVIMENTO</b>	<b>12</b>
3.1	Módulo 1	12
3.2	Módulo 2	13
3.3	Módulo 3	15
3.4	Interface	16
3.5	Comunicação entre os módulos	19
<b>4</b>	<b>RESULTADOS</b>	<b>21</b>
4.1	Módulo 1	21
4.2	Módulo 2	22
4.3	Módulo 3	23
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>25</b>
	<b>REFERÊNCIAS</b>	<b>26</b>

# 1 Introdução

Em uma edificação a energia é responsável por uma grande parte dos gastos e por esse motivo é interessante possuir um controle da utilização da mesma. Com essa necessidade surgiu o conceito de automação predial, que deriva do conceito de gerência de energia(EHRLICH, 2008)

Com a automação predial é possível conciliar um aumento da eficiência energética sem diminuir o conforto que é fortemente relacionado com a produtividade dos ocupantes. Ainda pode garantir a segurança podendo, por exemplo, detectar e avisar aos bombeiros incêndios em fase inicial e direcionar os ocupantes a saída mais apropriada.

Também é possível diminuir o custo de manutenção pois não haverá desperdícios na utilização de equipamentos e sistemas. Apagando as lâmpadas se ninguém estiver no ambiente resultará em um maior tempo sem que seja preciso a troca, com a troca de uma lâmpada deve ser levado em conta não só o valor da lâmpada, mas o custo da mão de obra de substituição e do seu descarte.

## 1.1 Objetivos

O trabalho de conclusão de curso proposto neste documento tem como objetivo a criação de um sistema capaz de controlar a temperatura, umidade e as luzes de uma sala.

## 1.2 Motivação

Em um local de trabalho, onde trabalham várias pessoas é relativamente fácil esquecer o ar condicionado ligado ou as luzes acesas quando todos foram embora, pois as vezes não se sabe se é o último a sair ou não, deixando assim o ar condicionado e luzes ligadas. Também é preferível tornar o ambiente com as melhores condições, pois assim aumentará a produtividade dos trabalhadores. Este trabalho cria um sistema capaz de resolver até um certo nível estes problemas.

## 1.3 Metodologia

Para a elaboração do sistema foi feito um estudo de tópicos sobre automação predial e maneiras de implementar um sistema para a automação predial

Em seguida foi criado um primeiro protótipo em protoboard do sistema e por fim foi feita a elaboração de protótipos montados em uma PCI.

## 1.4 Estrutura do documento

No Capítulo 2 é feita uma breve descrição dos componentes utilizados para a criação do sistema composto por três módulos.

No Capítulo 3 foi mostrado o desenvolvimento de cada modulo individualmente e seu funcionamento. Então no Capítulo 4 foi mostrado o resultado da criação dos módulos, expondo os protótipos criados.

O trabalho é finalizado no Capítulo 5 com o encaminhamento das conclusões e propostas de refinamentos para pesquisas similares posteriores.

## 2 Sistema

O sistema criado neste trabalho possui 3 módulos que em conjunto são capazes de realizar o controle da temperatura, umidade e iluminação da sala de uma forma inteligente.

O módulo 1 é o principal, responsável pelas decisões de controle, obtenção da temperatura e umidade da sala e a interação com o usuário através de um computador. O módulo 2 controla as luzes e o módulo 3 controla o ar condicionado, detecta se as luzes estão acesas e detecta movimento na sala.

Para a comunicação entre os módulos foi usado o protocolo MQTT que utiliza o padrão de troca de mensagens publicador/subscritor. Neste padrão, quando um elemento da rede deseja receber uma determinada informação, ele a subscrive, fazendo uma requisição para um outro elemento da rede capaz de gerir as publicações e subscrições.

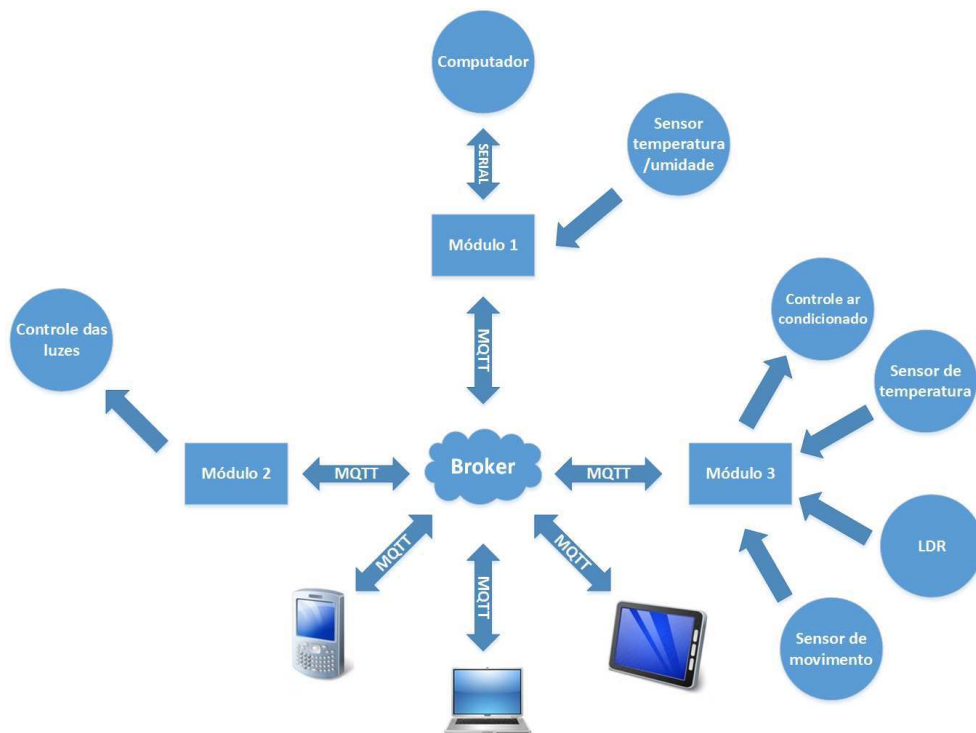


Figura 1 – Esquema do sistema

### 2.1 Especificação do Módulos

#### 2.1.1 Módulo 1

O módulo 1, que é controlado por um Arduino/Genuino MKR1000, é responsável por coletar informações dos outros módulos e, com base nestas informações, decidir quais



ações devem ser tomadas para manter a sala com os valores de temperatura, umidade e luminosidade o mais próximo do ideal. Ele também é responsável por desligar as luzes e o ar condicionado caso não tenha movimentação na sala.

Através de uma interface de comunicação serial entre o módulo e um computador, com sistema operacional Windows, é possível definir preferências de temperatura e umidade que são armazenados em um cartão microSD. Também é possível acessar data e hora de registros como temperatura e umidade máxima ou mínima, movimentos na sala entre outros.

### 2.1.1.1 Arduino/Genuino MKR1000

Arduino é uma plataforma de prototipagem eletrônica, criado por Massimo Banzi e David Cuartielles em 2005 com objetivo de permitir o desenvolvimento de controle de sistemas interativos, de baixo custo e acessível a todos. O projeto foi criado pensando em artistas e amadores, ou seja, não é necessário ter conhecimentos prévios em eletrônica ou programação para iniciar-se no mundo Arduino. Outra característica importante é que todo material (software, bibliotecas, hardware) é open-source, ou seja, pode ser reproduzido e usado por todos sem a necessidade de pagamento de royalties ou direitos autorais.

O Arduino/Genuino MKR1000 foi criado com o objetivo de facilitar o desenvolvimento de aplicações que necessitam de conectividade WI-FI pois é uma placa de fácil uso e com custo acessível. Possui um microcontrolador ARM resultando em um grande poder de processamento, ideal para controlar o módulo principal do sistema criado neste trabalho. Possui um carregador de bateria Li-Po, sendo possível carregar a bateria enquanto está sendo alimentada por uma fonte externa. A mudança entre as fontes é feita automaticamente e pode funcionar com ou sem a bateria.

Tabela 1 – Característica do MKR1000

Variável	Descrição
Microcontrolador	SAMD21 Cortex-M0+ 32bit low power ARM MCU
Tensão de entrada (USB/VIN)	5 V
Bateria suportada	Uma célula Li-Po, 3.7 V, mínimo 700mAh
Tensão de operação	3.3 V
Pinos I/O digitais	8
Pinos PWM	12 (0, 1, 2, 3, 4, 5, 6, 7, 8, 10, A3, A4)
UART	1
SPI	1
I2C	1
Pinos de entrada analógica	7 (ADC 8/10/12 bit)
Pinos de saída analógica	1 (DAC 10 bit)
Interrupções externas	8 (0, 1, 4, 5, 6, 7, 8, A1, A2)
Corrente DC por pino I/O	7 mA
SRAM	32 KB
Flash Memory	256 KB
EEPROM	Não
Velocidade do clock	32.768 kHz (RTC), 48 MHz

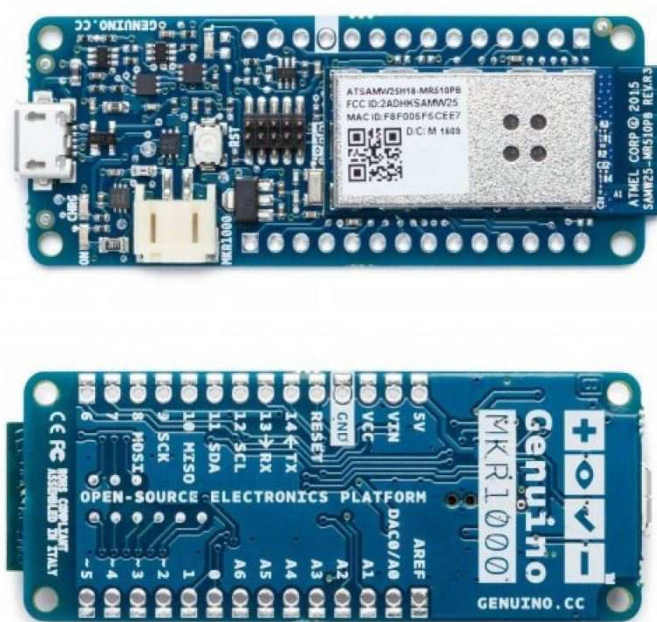


Figura 2 – Arduino/Genuino MKR1000

### 2.1.1.2 Sensor de Temperatura e Umidade RHT03

É um sensor de baixo custo e fácil utilização que realiza medições imediatas de umidade relativa e temperatura utilizando apenas uma única saída, o pino 2. Esta saída é conectada a uma entrada digital do microcontrolador com um resistor de 1 k Ohm de Pull-Up. O pino 1 é conectado a uma tensão de 3.3 V e o pino 4 conecta em Gnd. O pino

3 não é conectado a nada.

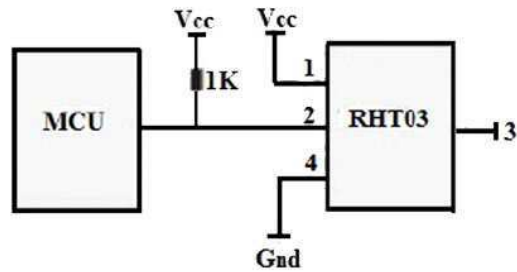


Figura 3 – Esquema de conexões do RHT03

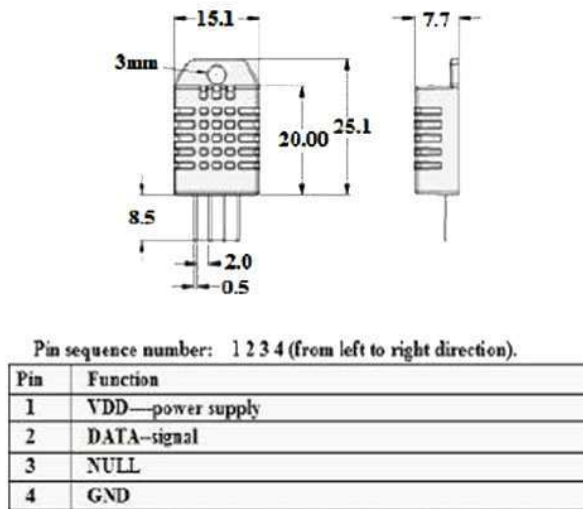


Figura 4 – Pinagem do RHT03

### 2.1.1.3 LDR

O LDR é um componente cuja resistência varia de acordo com a intensidade da luz. Quanto mais luz incidir sobre o componente, menor a resistência.

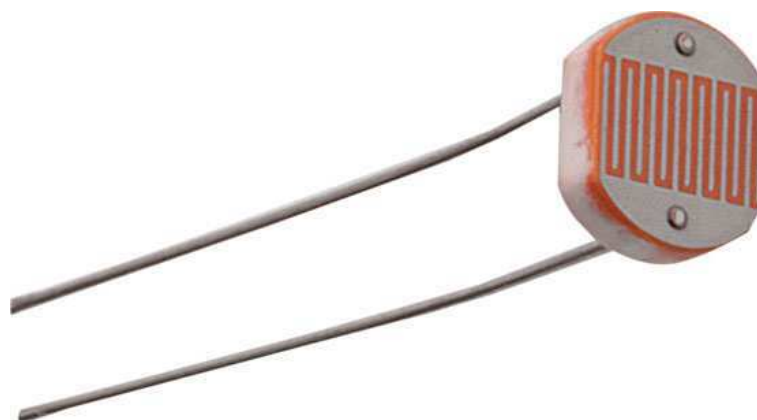


Figura 5 – LDR

Sua ligação no microcontrolador é feita através do seguinte esquema:

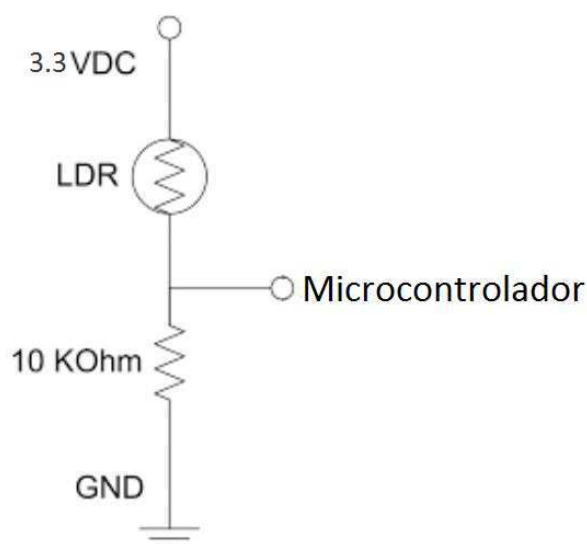


Figura 6 – Esquema de ligação do LDR ao microcontrolador

#### 2.1.1.4 Módulo Cartão SD Card

Este módulo permite a leitura e escrita em um cartão MicroSD, suporta formatos de arquivo FAT16 e FAT32. A alimentação é compatível com o Arduino/Genuino MKR1000, 3.3 V e a comunicação com o microcontrolador é feita através de uma interface SPI (ARDUINO, ) por meio dos pinos MOSI, SCK, MISO e CS.



Figura 7 – Módulo Cartão SD Card

#### 2.1.1.5 Módulo relé

Este módulo é capaz de acionar cargas de 220V AC, como lâmpadas, equipamentos eletrônicos, motores. O controle é feito através de um sinal positivo de 5V, como o sinal da saída de um controlador, e a alimentação é uma tensão de 5V. O módulo garante o isolamento do lado de tensão 220V com o lado do microcontrolador utilizando um fotoacoplador.



Figura 8 – Módulo Relé

### 2.1.2 Módulos 2 e 3

A família de módulos que utilizam o chip ESP8266 é uma ótima alternativa para um projeto IoT, pelo tamanho, praticidade, baixo consumo de energia e custo. Para controlar o Módulo 2 e 3 foi utilizado o ESP8266 ESP-12F e ESP8266 ESP-01 respectivamente

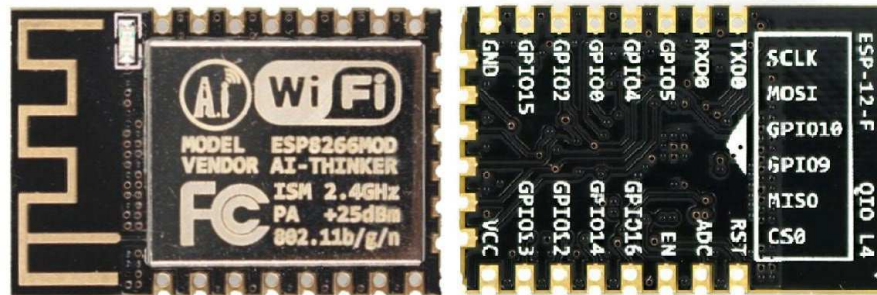


Figura 9 – Módulo ESP8266 ESP-12F

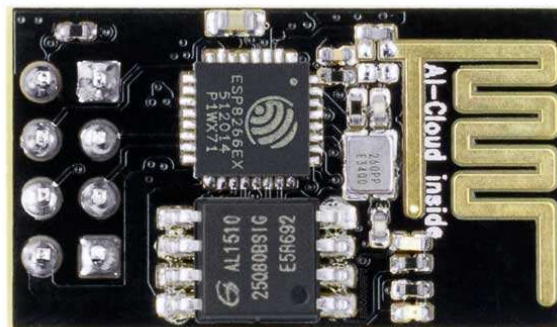


Figura 10 – Módulo ESP8266 ESP-01

Estes controladores possuem suporte para WiFi 802.11 b/g/n e já conta com uma antena integrada a placa. Os módulos ESP8266 possuem conectores GPIO, barramentos I2C, SPI, UART, entrada ADC, saída PWM e sensor interno de temperatura uma CPU que opera em 80MHz, com possibilidade de operar em 160MHz. Possuem também além da memória RAM e ROM uma memória flash de 512KB. Possuem uma tensão de funcionamento de 3.3 V e foi utilizado a IDE do Arduino para a programação.

### 2.1.3 Adaptador USB serial TTL

Para programar os microcontroladores ESP8266 ESP-12F e ESP8266 ESP-01 foi necessário o uso de um adaptador, pois estes não possuem conexão USB sendo necessário

o uso do adaptador para fazer a ligação entre o microcontrolador e o computador. A conexão é feita através dos pinos RX e TX dos controladores e no computador USB.

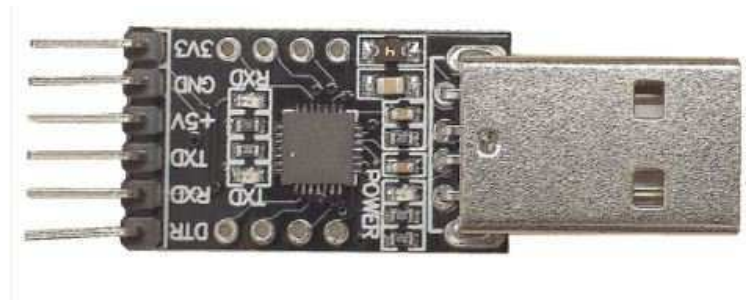


Figura 11 – Adaptador USB serial TTL

#### 2.1.4 Protocolo MQTT

MQTT é um protocolo aberto para M2M (Machine-to-Machine) projetado para um baixo consumo de banda de rede e requisitos de hardware sendo extremamente simples e leve. O MQTT foi desenvolvido pela IBM e Eurotech e é projetado para enviar dados através de redes com baixa banda de dados, para isto o protocolo é desenvolvido em cima de vários conceitos que garantem uma alta taxa de entrega das mensagens. O MQTT permite comunicação bidirecional utilizando um padrão publicação/subscrição para a troca de mensagens. É um protocolo bastante utilizado em aplicações IoT pela sua simplicidade e facilidade de implementação.

Uma comunicação MQTT é composta de clientes, que podem ser publicadores, subscritores ou publicadores e subscritores, e um broker que é responsável por gerir as trocas de mensagens. O protocolo MQTT é baseado no TCP/IP e ambos, cliente e broker, necessitam da pilha TCP/IP para o seu funcionamento. O MQTT está na mesma camada OSI que o HTTP, porém a maior diferença entre os dois protocolos é o tamanho do payload. No HTTP, o payload é maior, o que inviabiliza o seu uso em conexões de baixa qualidade, como GSM por exemplo. O protocolo utiliza criptografia (SSL/TLS) para utilizar a opções de usuário e senha.

O cliente inicia a conexão com o broker através de uma mensagem de conexão contendo o seguinte conteúdo:

- clientId
- cleanSession
- username
- password
- keepAlive



Na opção de `clientId` é um nome para identificação que todo cliente deve ter, a `cleanSession` é uma flag que indica se o cliente quer uma conexão persistente ou não, com uma conexão persistente o broker irá guardar todas as mensagens que o cliente está subscrito até que essas mensagens sejam recebidas pelo cliente. Em `username/password` é enviado os dados de autenticação do cliente. O cliente envia um ping em tempos regulares para avisar ao broker que ele ainda está conectado e em resposta o broker envia um ping também, a regularidade de tempo que isto ocorre é definido em `keepalive`.

Quando o broker recebe a mensagem de conexão ele responde ao cliente com uma mensagem contendo `Session Present` flag e `Connect acknowledge` flag. Na flag `Session Present` o broker indica se o cliente já possui uma conexão com ele e na flag `Connect acknowledge` o broker informa se a conexão foi feita e se caso não for feita é enviado qual foi o problema.



Figura 12 – Esquema de uma conexão MQTT

As mensagens são feitas em formato de tópicos:

**Tópico/payload**

Em que “payload” é a informação da mensagem e “Tópico” é a identificação da informação da mensagem. A identificação da informação pode ter mais de um nível:

**Tópico/1/payload**

**Tópico/2/payload**

Assim um cliente poderá subscrever apenas na mensagem “Tópico/1” ou na mensagem “Tópico/2”. Um cliente também pode subscrever na mensagem “Tópico/+” para receber as duas informações agrupadas.



## 3 Desenvolvimento

Para o desenvolvimento dos módulos foi utilizado um protoboard para fazer as devidas ligações dos componentes e microcontroladores. Foi utilizado a linguagem de programação Arduino, que consiste em uma linguagem baseada na linguagem Wiring, utilizando o ambiente do Arduino para a programação.

### 3.1 Módulo 1

Existe seis opções para o controle feito pelo módulo 1, ativar ou desativar o controle automático da temperatura, da luz ou da umidade, a temperatura ideal da sala, a umidade ideal da sala e o tempo máximo sem movimentação dentro da sala. Estas opções ficam salvas em um SD card não sendo necessário configura-las novamente caso ocorra um desligamento inesperado do módulo, este carrega estas opções assim que for ligado. É possível alterar estas opções através da interface do módulo 1 com o computador.

Inicialmente é verificado se existe conexão com a rede WIFI, caso não exista será tentado a conexão até obter êxito, caso exista conexão será feito o loop do cliente que consiste em uma função que verifica se existe mensagens chegando originada do broker relacionada aos tópicos que o microcontrolador, como um cliente, fez a subscrição. Em seguida é feita a leitura da temperatura e umidade da sala pelo sensor RHT03 e então é feita uma verificação para saber se as leituras foram feitas corretamente, caso exista algum erro na leitura do sensor ele fará as leituras novamente. Com as leituras da temperatura e umidade feitas corretamente é verificado se algum comando originado da interface entre o módulo 1 e um computador foi enviado, caso exista algum comando ele é executado.

Não havendo nenhum comando ou após a execução do comando o módulo 1 verificará se existe movimentação dentro da sala e caso não exista será verificado se o tempo sem movimento é maior que o tempo definido pelo usuário. Se o tempo sem movimentação for maior que o tempo definido pelo usuário é verificado se o ar condicionado está ligado e estando ligado ele será desligado, em seguida o mesmo processo é feito com as luzes.

Se o tempo sem movimentação for menor que o definido pelo usuário, ou houver movimentação na sala, é verificado se o controle do ar condicionado está ligado, caso esteja é garantido que o ar condicionado esteja ligado com uma temperatura especificada pelo usuário e em sequência o mesmo procedimento é feito para o controle das luzes.

Em seguida é verificado se o controle de umidade está ligado, caso esteja o sistema irá garantir uma umidade igual a escolhida pelo usuário. O controle da umidade é feito utilizando dois equipamentos, o ar condicionado e um umidificador de ar. Para

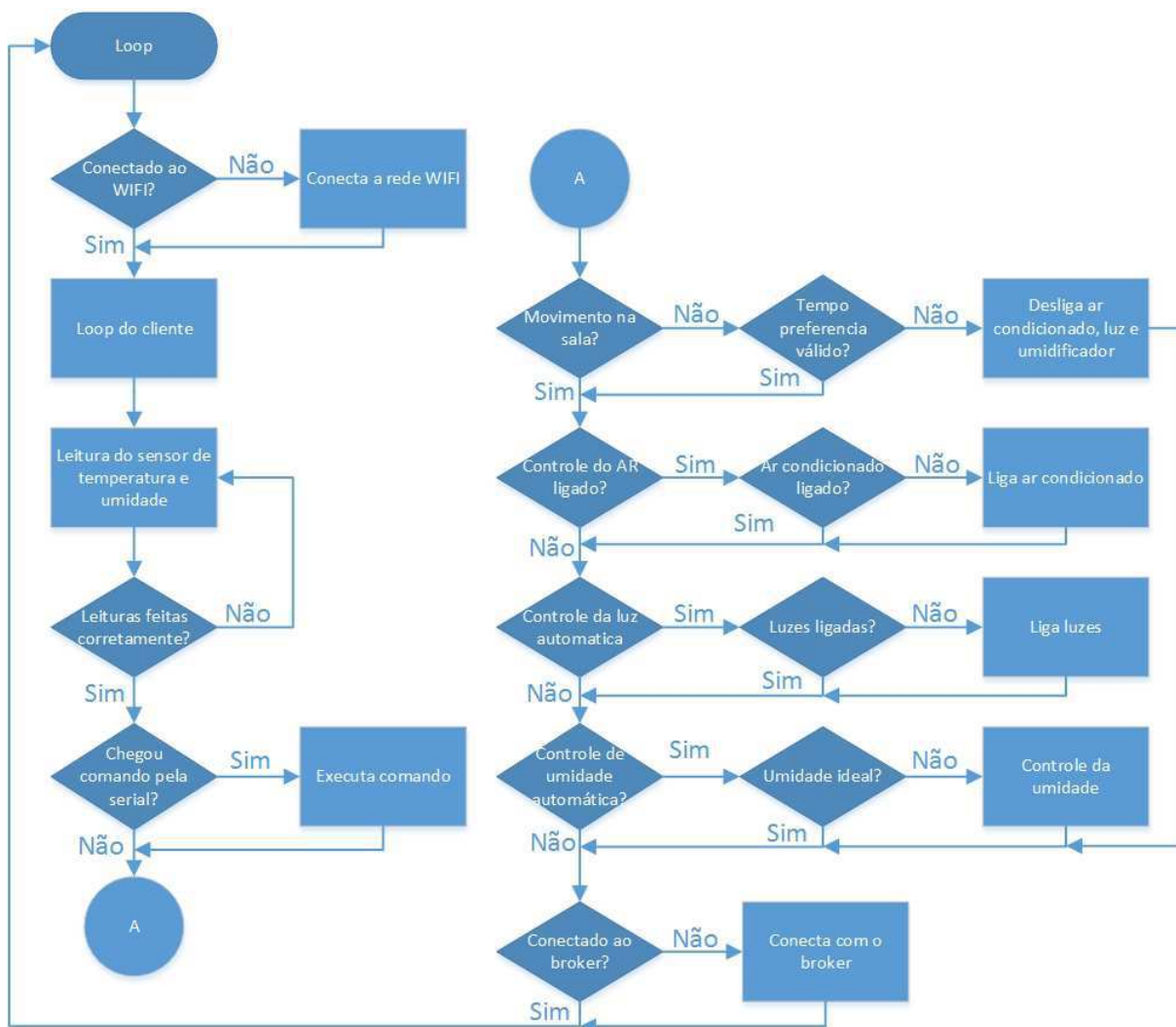


Figura 13 – Fluxograma da lógica do módulo 1

desumidificar o ar é utilizado o ar condicionado que tem a função de desumidificar o ar e para umidificar o ar é utilizado um umidificador de ar. No sistema construído neste trabalho foi implementando o controle de um quarto módulo muito semelhante ao módulo 2, responsável pelas luzes, porém não foi criado um protótipo.

Por fim é verificado se existe conexão com o broker e na hipótese de não existir conexão é tentando uma conexão com o broker. Em seguida começa novamente o loop.

### 3.2 Módulo 2

No loop feito pelo módulo 2 é verificado se existe conexão com a rede WIFI, caso não exista será tentado a conexão até obter êxito e quando houver conexão será feito a loop do cliente que consiste em uma função que verifica se existe alguma mensagem dos tópicos que foi feito a subscrição.

Após o loop do cliente o módulo 2 verifica, caso tenha recebido alguma mensagem,

se deve ligar ou desligar as luzes caso receba uma mensagem pedindo para ligar ou desligar as luzes o módulo dois executa esse pedido, por fim é verificado se existe conexão com o broker e caso não exista é tentado uma conexão até obter êxito e havendo êxito o loop recomeça.

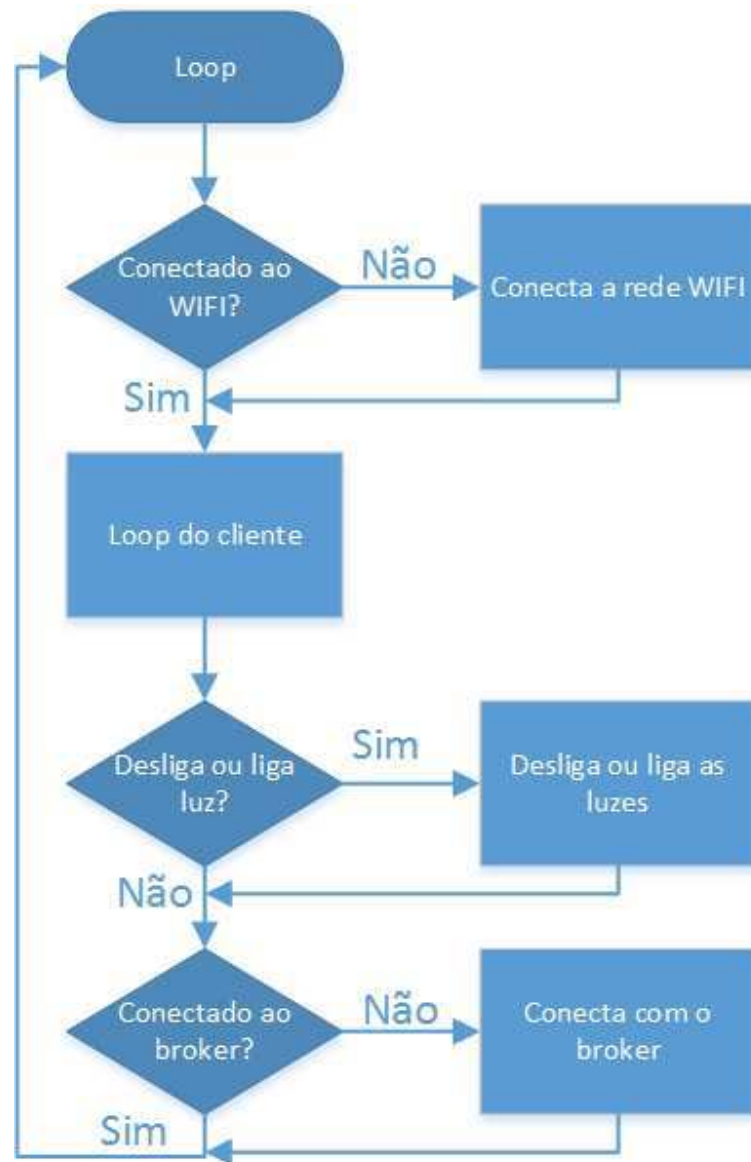


Figura 14 – Fluxograma da lógica do módulo 2

O módulo 2 tem a simples função de desligar ou ligar as luzes, porém é importante manter a opção de acender ou apagar as luzes usando o interruptor, para isso é utilizado uma ligação Three Way que consiste em um tipo de ligação capaz de acender a luz em um interruptor e apaga-la em outro. No caso, em vez de outro interruptor será usado o modulo 2, permitindo assim ligar e desligar as luzes com o interruptor ou com o modulo 2.

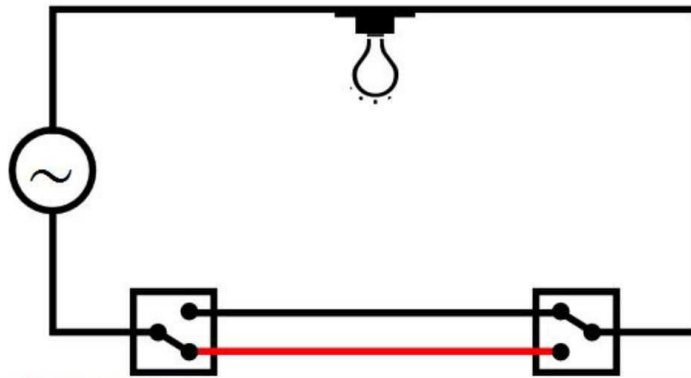


Figura 15 – Ligação three way

### 3.3 Módulo 3

Uma das funções do módulo 3 é enviar o comando infravermelho, comando infravermelho que é enviado ao ar condicionado é uma mensagem codificada invisível ao olho humano, esta mensagem é enviada através de pulsos do led infravermelho resultando em um código binário. Este código tem quatro partes, a primeira é para informar que está no início do comando, a segunda é referente ao comando, a terceira é o endereço do aparelho que se destina o código e a última parte é para informar o fim do comando.



Figura 16 – Código infra vermelho

Foi utilizado o Arduino/Genuino em conjunto com um receptor infravermelho para obter os códigos de controle do ar condicionado.

Outra função é detectar se existe movimentação na sala, necessitando assim ficar em um ponto que consiga detectar a movimentação na maior parte da sala. Também é necessário ficar próximo as luzes para saber se as luzes estão ligadas ou a iluminação é natural. Com base nestas informações foi escolhido o teto como o melhor lugar para o módulo 3.

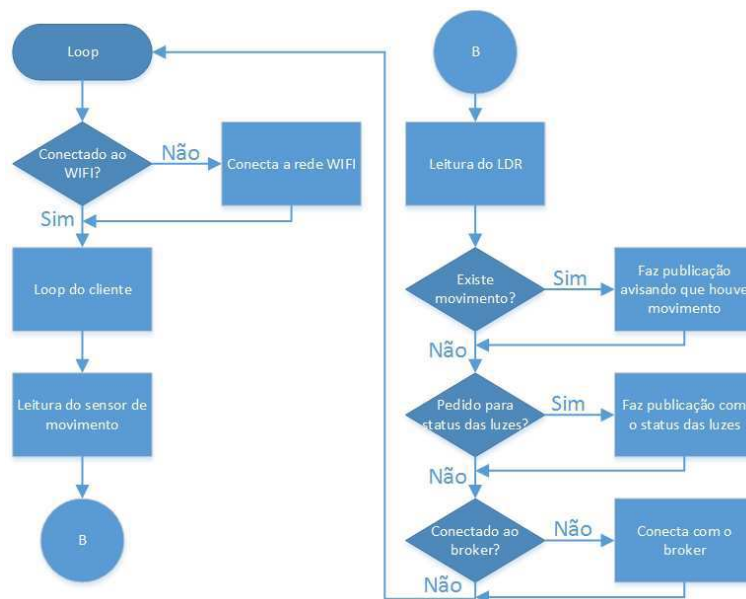


Figura 17 – Fluxograma da lógica do módulo 3

Assim como os outros dois módulos, é feita a confirmação da existência de uma conexão WiFi e após esta confirmação entra-se no loop do cliente, no loop do cliente o módulo 3 irá verificar se existe alguma publicação relacionado aos tópicos subscritos. Após o loop do cliente é feita a leitura do sensor de movimento e em seguida a do sensor LDR, após as leituras é verificado se ocorreu movimento na sala e caso haja movimentação uma publicação é feita relatando a movimentação. Em seguida é verificado se o módulo 1 fez um pedido, pedido que é feito através das publicações que o módulo 3 é subscrito, para saber o status das luzes e caso seja verificado o pedido é feita uma publicação informando o status das luzes. Por fim é feita a confirmação de conexão com o broker, processo semelhante aos outros dois módulos e havendo conexão o loop recomeça.

### 3.4 Interface

Para a interface de comunicação com o computador foi criado um aplicativo Windows Forms através da IDE do Visual Studio usando a plataforma .NET Framework com uma linguagem C#. Esta escolha se deu por ser um aplicativo de fácil implementação que atende as necessidades do sistema.

A interface é dividida em quatro abas, em que a primeira aba, chamada de Conexão com módulo, é responsável por fazer a conexão com o módulo 1, basta escolher a porta COM que o módulo está conectado com o computador e clicar em conectar.

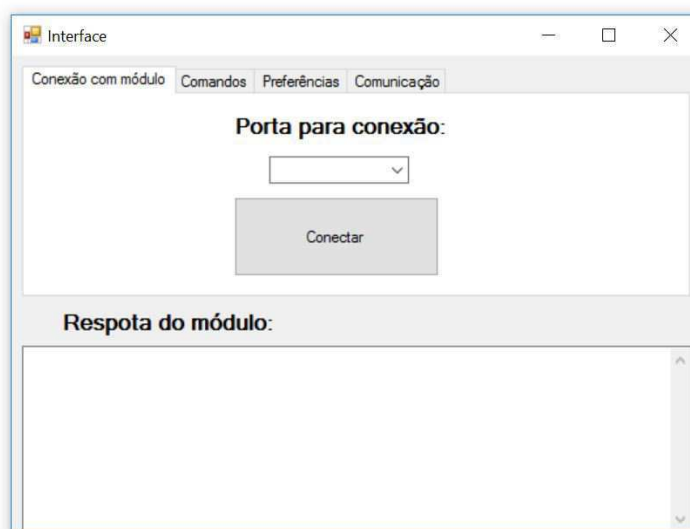


Figura 18 – Aba Conexão com módulo

A segunda aba, chamada Comandos, possui 6 botões:

- Ligar luzes
- Ligar umidificador
- Desumidificar
- Ligar ar condicionado
- Umidade
- Temperatura

O botão Ligar luzes pode ligar ou desligar as luzes, o botão Ligar umidificador liga o umidificador, o botão Desumidificar aciona o modo de desumidificar do ar condicionado, o botão Ligar ar condicionado liga ou desliga o ar condicionado, o botão Umidade informa a umidade da sala e o botão Temperatura informa a temperatura da sala. A aba ainda possui um espaço que mostrará a resposta enviada pelo módulo.

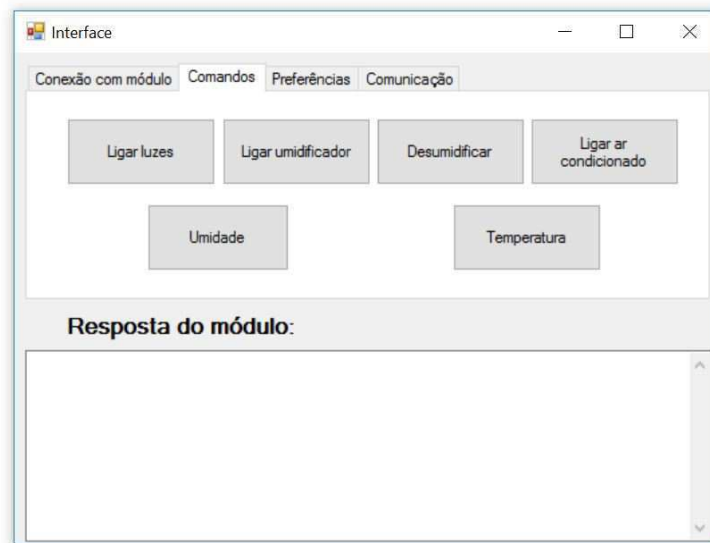


Figura 19 – Aba Comandos

A terceira aba, chamada Preferências, possui 9 botões:

- Tempo +
- Tempo -
- Umidade +
- Umidade -
- Temperatura +
- Temperatura -
- Ar condicionado automático
- Umidade automática
- Luzes automáticas

Com os botões Tempo + e Tempo - é possível fazer o ajuste do tempo máximo que não é detectado movimentação na sala sem que o ar condicionado, as luzes e o umidificador sejam desligados. Com os botões Umidade +, Umidade -, Temperatura + e Temperatura - é possível fazer o ajuste da umidade e temperatura, e os botões Ar condicionado automático, Umidade automática e Luzes automática habilitam o controle automático do ar condicionado, umidade e luzes respectivamente.

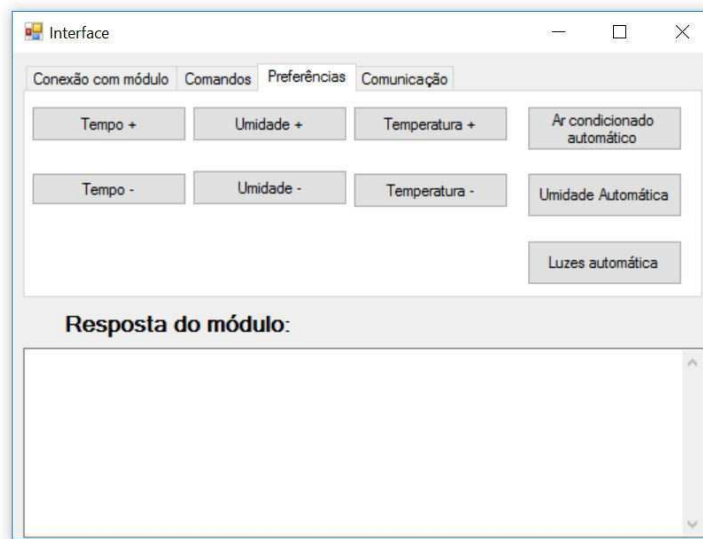


Figura 20 – Aba Preferências

E na última aba, chamada Comunicação, é possível enviar comandos para o módulo para saber data e hora dos eventos ocorridos na sala, como por exemplo hora e data do desligamento das luzes e do ar condicionado por não haver ninguém na sala.

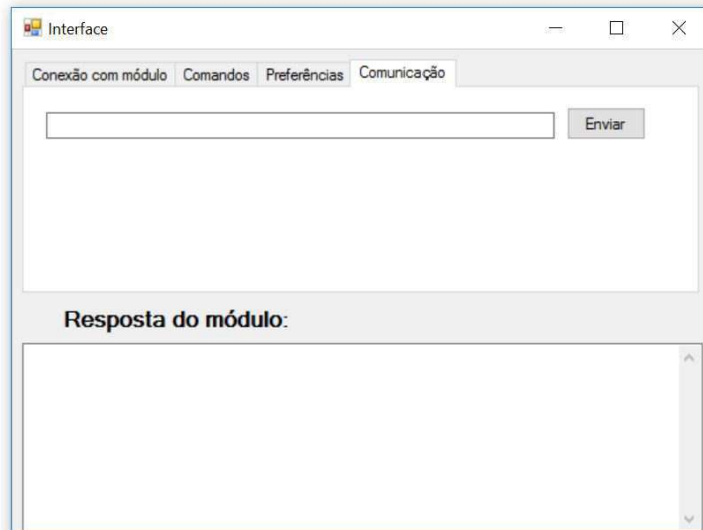


Figura 21 – Aba Comunicação

### 3.5 Comunicação entre os módulos

Para a implementação do protocolo de comunicação escolhido, o MQTT, foi utilizado como broker o site <https://shiftr.io> que faz o papel do broker gratuitamente, não sendo necessário a implementação de um broker local.



Uma vantagem de escolher um broker online é a possibilidade de conectar dispositivos móveis como celular ou tablete onde eles podem tanto se inscrever em tópicos quanto publicar tópicos.

A rede MQTT para este sistema pode ser representado pela seguinte imagem:

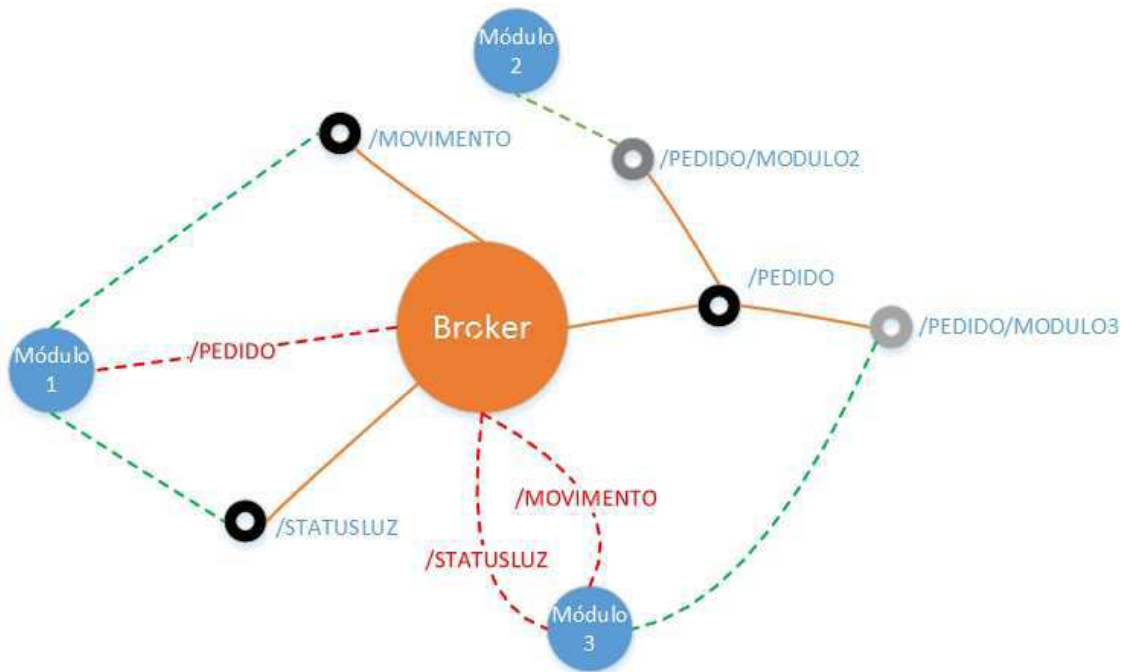


Figura 22 – Rede MQTT

No centro desta imagem está o broker em laranja, os tópicos (/PEDIDO/MOVIMENTO e /STATUSLUZ) de preto e de cinza os subtópicos (/PEDIDO/MODULO2 e /PEDIDO/MODULO3). Os clientes que são os três módulos estão representados de azul ligado por linhas tracejadas verdes as suas respectivas inscrições e saindo dos módulos, representado por linhas tracejadas vermelhas, suas respectivas subscrições.

Através dessa comunicação o módulo um faz os devidos pedidos. Para o módulo dois essa comunicação é feita para pedir que ele acenda ou apague as luzes e para o módulo três é feito os pedidos de envio dos códigos através do emissor infravermelho para o ar condicionado. Como esta é uma comunicação bidirecional, o módulo 3 envia informações sobre a movimentação na sala e se as luzes estão acesas.

## 4 Resultados

Após o desenvolvimento utilizando a protoboard foi desenvolvido um protótipo para cada módulo e cada protótipo foi montado em uma PCI.

### 4.1 Módulo 1

Para o protótipo do modulo 1 foi utilizado um leitor de SD card e um sensor RHT03, como o módulo 1 ficará conectado a um computador ele não precisaria de uma fonte externa pois poderia ser alimentando através da conexão USB com o computador e no caso do computador desligar o controlador Arduino/Genuino MKR1000 possui uma entrada para bateria de lítio e um gerenciador de carga para a bateria, porém para que o módulo 1 funcione mesmo que não exista um computador foi adicionado uma entrada para uma fonte externa de 5V.

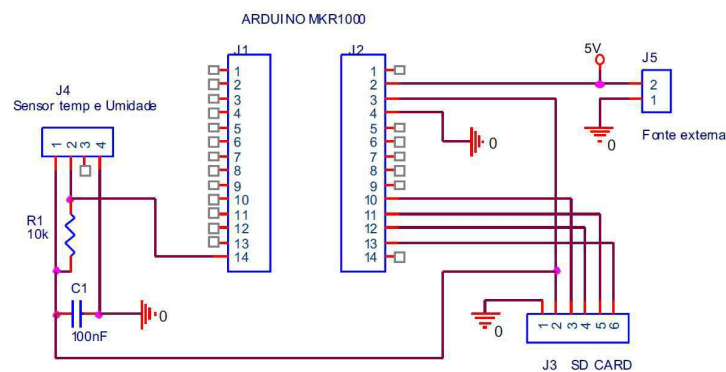


Figura 23 – Esquema do módulo 1



Figura 24 – Módulo 1

## 4.2 Módulo 2

O protótipo do módulo 2 utilizou um relé para ligar e desligar as luzes e necessita de uma entrada de 5V, 3.3V e Gnd.

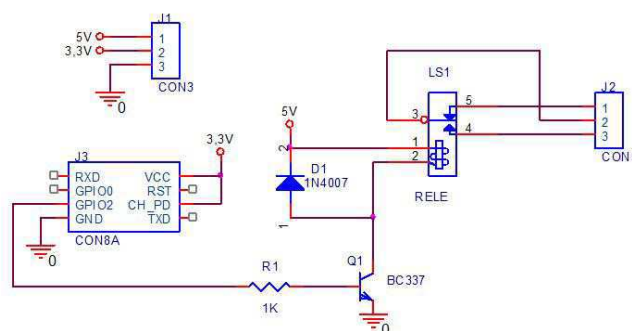


Figura 25 – Esquema do módulo 2



Figura 26 – Módulo 2

### 4.3 Módulo 3

O protótipo do módulo 3 possui sensor de presença, emissor infravermelho e o LDR onde os três sensores podem ser desconectados do módulo e adicionado uma extensão para que fique distante do módulo. Apresentou bons resultados quando foi posicionado no teto da sala e foi adicionado dois cabos, um para deixar o emissor infravermelho mais próximo do receptor infravermelho do ar condicionado e o outro cabo para deixar o LDR perto das luzes.

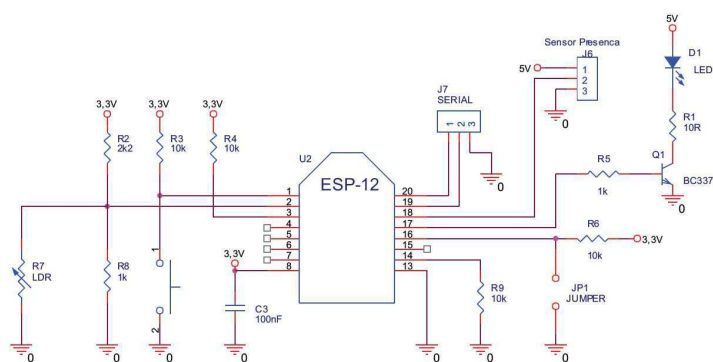


Figura 27 – Esquema do módulo 3

O protótipo possui um regulador 3.3V, regulando a tensão de entrada 5V para 3.3V sendo necessário a entrada de apenas uma tensão de 5V.

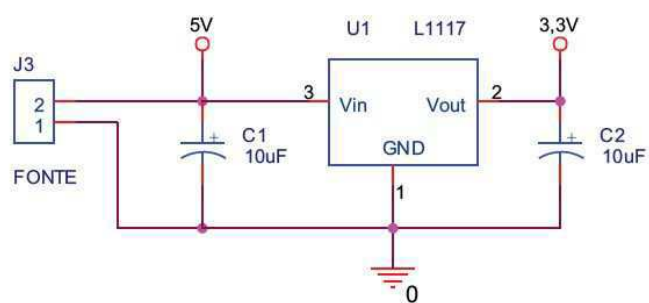


Figura 28 – Regulador 3.3V

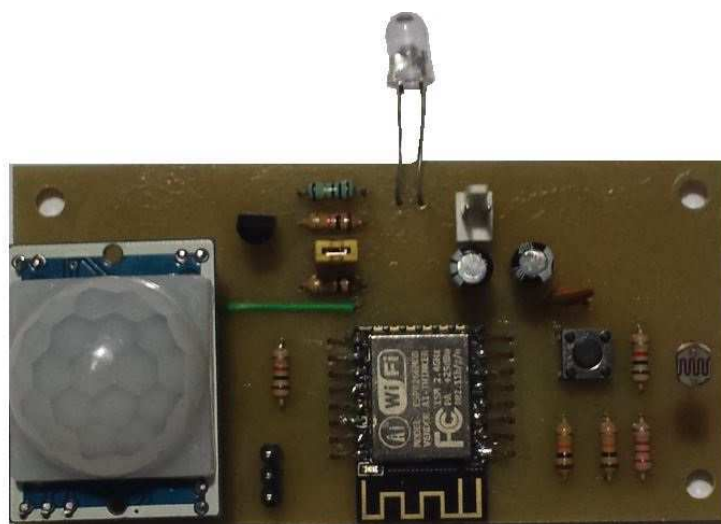


Figura 29 – Módulo 3

## 5 Considerações Finais

Obteve-se sucesso com o controle da temperatura, umidade e luzes da sala, o sistema mantém a sala na temperatura e umidade próximas a desejada pelo usuário e assegura que não seja desperdiçada energia desligando luzes e ar condicionado caso não possua ninguém na sala.

Como proposta de refinamento para pesquisas similares posteriores, tem-se a implementação de uma iluminação com LED que possa ser variada de acordo com as necessidades do local, um controle das cortinas para controlar a iluminação natural e um controle de acesso a sala, onde só pessoas autorizadas podem entrar na sala.

# Referências

ARDUINO, S. A brief introduction to the serial peripheral interface (spi). Disponível em: <<https://www.arduino.cc/en/Reference/SPI>>. Citado na página 7.

EHRlich, P. Green intelligent buildings – a brief history. 2008. Disponível em: <<http://www.automatedbuildings.com/news/mar08/columns/080225045606erhlich.htm>>. Citado na página 1.