



CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA



Universidade Federal
de Campina Grande

ARTUR SOUSA FREITAS

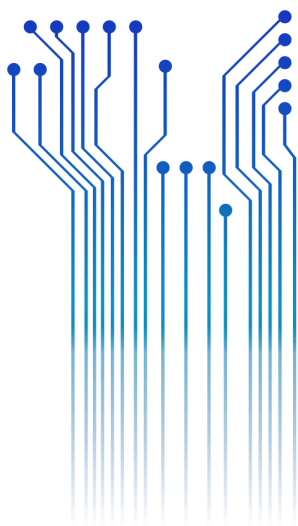


Centro de Engenharia
Elétrica e Informática

RELATÓRIO DE ESTÁGIO SUPERVISIONADO
SMARTIKS TECNOLOGIA DA INFORMAÇÃO



Departamento de
Engenharia Elétrica



Campina Grande, Paraíba
Abril de 2017

ARTUR SOUSA FREITAS

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

*Relatório de Estágio Supervisionado submetido
à Unidade Acadêmica de Engenharia Elétrica
da Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Sistemas Embarcados

Orientador:

Francisco das Chagas Fernandes Guerra, D.Sc

Campina Grande, Paraíba
Abril de 2017

ARTUR SOUSA FREITAS

RELATÓRIO DE ESTÁGIO SUPERVISIONADO

*Relatório de Estágio Supervisionado submetido
à Unidade Acadêmica de Engenharia Elétrica
da Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Sistemas Embarcados

Aprovado em / /

Professor Avaliador
Universidade Federal de Campina Grande
Avaliador

Francisco das Chagas Fernandes Guerra, D.Sc.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho aos meus pais e irmãos,
por todo o apoio e incentivo durante minha vida
acadêmica.

AGRADECIMENTOS

Agradeço a Deus, em primeiro lugar, pela minha vida, pela minha saúde e pelas oportunidades cujo me foram concedidas.

Agradeço também à meus pais João e Arleide Freitas por terem se esforçado tanto para me proporcionar uma boa educação, por terem me alimentado com saúde, força e coragem, as quais foram essenciais para superação de todas as adversidades ao longo desta caminhada.

Agradeço também aos meus irmãos Raul e Rafael, por todo o incentivo e suporte durante minha vida acadêmica.

Agradeço ao Professor Francisco das Chagas, por ter aceitado me orientar neste estágio, a Eloi Rocha pela oportunidade de estagiar na Smartiks e a Plateny Ponchet e Victor Perone pela assistência durante o período de estágio.

Enfim, agradeço a todos que de alguma forma, passaram pela minha vida e contribuíram para a construção de quem sou hoje.

*“Se você quiser descobrir os segredos do Universo , pense em termos de energia,
frequência e vibração”*

Nikola Tesla

RESUMO

Neste documento estão descritas as atividades desenvolvidas pelo aluno Artur Sousa Freitas durante a realização do estágio supervisionado com carga horária de 330 horas na empresa SMARTIKS TECNOLOGIA DA INFORMAÇÃO LTDA-ME, localizada em Campina Grande – PB , no período de 28 de novembro de 2016 a 15 de março de 2017, sob a orientação do professor Francisco das Chagas Fernandes Guerra.

Durante o estágio, foi possível: acompanhar as etapas finais de desenvolvimento de um *smart meter* trifásico, elaborar um manual de instalação para medidores monofásicos e trifásicos inteligentes, efetuar o *upload* de *bootloader* e *firmwares* nos microcontroladores utilizados pela empresa, acompanhar a instalação dos medidores e projetar um sistema embarcado capaz de controlar o ar condicionado da empresa por meio da internet utilizando o protocolo de comunicação MQTT.

Palavras-chave: Smartiks, Sistemas Embarcados, Internet das Coisas, Medidores Inteligentes, MQTT.

ABSTRACT

In this document are described the activities performed by the student Artur Sousa Freitas during the realization of his supervised internship at the company SMARTIKS TECNOLOGIA DA INFORMAÇÃO LTDA-ME, located at Campina Grande-PB, from November 28 of 2016 to March 15 of 2017, under Professor Francisco das Chagas Fernandes Guerra orientation.

During the internship, it was possible to: follow the final development stages of a three-phase smart meter, elaborate an installation manual for smart meters, make the firmware upload on the microcontrollers used in the meters built by the company, install smart meters and develop an embedded system capable of control the air conditioner on the hardware room of the company with the internet by using the MQTT message protocol.

Keywords: Smartiks, embedded systems, Internet of Things, Smart Meters, MQTT.

LISTA DE ILUSTRAÇÕES

FIGURA 1: FUNDAÇÃO PARQUE TECNOLÓGICO DA PARARAÍBA	13
Figura 2: TRIÂNGULO de Potências	17
Figura 3: DIAGRAMA DE BLOCOS DO ATMEGA328	18
Figura 4: PMIC AM90336A	19
Figura 5: MQTT - Padrão <i>publish/subscribe</i>	21
Figura 6: MEDIDORES MONOFÁSICOS INTELIGENTES INSTALADOS	23
Figura 7: Medidor Inteligente enviando dados para o servidor	23
Figura 8: MICROCONTROLADOR ATMEGA328	25
Figura 9: ADAPTADOR SOQUETE TQFP32	26
Figura 10: ESP8266 ENCAPSULADO COMO ESP-12	27
Figura 11: Led emissor infravermelho	28
Figura 12: SENSOR DE TEMPERATURA LM35.....	28
Figura 13: PROTÓTIPO DO PROJETO.....	29
Figura 14: VERSÃO FINAL DO PROJETO	29
Figura 15: INTERFACE WEB.....	30

LISTA DE ABREVIATURAS E SIGLAS

P	Potência Ativa
Q	Potência Reativa
fp	Fator de Potência
RMS	<i>Root Mean Square</i>
LED	<i>Light Emissor Diode</i>
RGB	<i>Red Green Blue</i>
SMD	<i>Surface-Mount Technology</i>
IoT	<i>Internet of Things</i>

SUMÁRIO

1	Introdução.....	12
1.1	Objetivos.....	12
1.2	Local do Estágio	13
1.3	Estrutura do Trabalho	14
2	Fundamentação teórica.....	15
2.1	Potência Elétrica em regime permanente senoidal.....	15
2.2	Microncontroladores	17
2.3	<i>Power Management Integrated Circuit</i> – PMIC.....	19
2.4	Internet das Coisas	20
2.4.1	Protocolo de comunicação MQTT.....	21
3	Atividades Desenvolvidas	22
3.1	Acompanhamento na Instalação dos medidores	22
3.2	Manual de Instalação medidores inteligentes.....	24
3.3	Upload de <i>firmware</i> em microcontroladores.....	24
3.4	Desenvolvimento de um sistema embarcado para controle de temperatura via internet.....	26
4	Conclusão	31
4.1	Considerações finais	31
4.2	Perspectivas futuras	31
	Referências	32
	APÊNDICE A – Código fonte do Projeto desenvolvido na atividade 4.....	33

1 INTRODUÇÃO

Atualmente, o mercado em torno do conceito Internet das Coisas se encontra em grande ascensão. Um dos segmentos que mais vem recebendo atenção nesta área é o de Redes Inteligentes (do inglês, *Smart Grid*). De forma básica, o termo *Smart Grid* refere-se a um sistema de energia elétrica que se utiliza da tecnologia da informação para fazer com que o sistema seja mais eficiente (econômica e energeticamente), confiável e sustentável.

Durante o estágio supervisionado na empresa SMARTIKS TECNOLOGIA DA INFORMAÇÃO LTDA-ME foi possível acompanhar de perto o desenvolvimento e teste de dispositivos capazes de coletar dados importantes da rede elétrica de uma instalação, e enviá-los por meio da internet para uma futura análise.

Durante a graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, o aluno recebe uma grande carga horária de aulas teóricas e práticas que dão ao aluno a base necessária para desempenhar atividades no ramo. O estágio supervisionado em Engenharia Elétrica é uma ótima oportunidade para aplicar o conhecimento adquirido durante os anos de graduação e é de suma importância para a formação do engenheiro eletricitista.

1.1 OBJETIVOS

O objetivo geral deste relatório é descrever as atividades desempenhadas pelo aluno Artur Sousa Freitas durante seu estágio supervisionado na empresa SMARTIKS TECNOLOGIA DA INFORMAÇÃO LTDA-ME, sob orientação do professor Francisco das Chagas Fernandes Guerra e supervisão do Engenheiro Eletrônico Plateny de Brito Ponchet, no período de 28 de novembro de 2016 a 15 de março de 2017.

O estágio se deu no setor de *hardware* da empresa e teve os seguintes objetivos específicos:

- Elaboração de um manual de instalação para medidores monofásicos e trifásicos inteligentes;
- Acompanhamento na instalação dos medidores monofásicos;

- Upload de firmware nos microcontroladores utilizados nos medidores desenvolvidos pela empresa;
- Desenvolvimento de um sistema embarcado para controlar a temperatura na sala de projetos da empresa por meio da internet utilizando o protocolo de comunicação MQTT;

1.2 LOCAL DO ESTÁGIO

FIGURA 1: FUNDAÇÃO PARQUE TECNOLÓGICO DA PARARAÍBA



Fonte: Elaborada pelo autor

O estágio ocorreu na sede de hardware da empresa, mais precisamente na Fundação Parque Tecnológico da Paraíba (PaqTcPB) mostrada na figura acima. A Smartiks foi criada a partir de um resultado de um projeto de pesquisa com a Chesf. Desenvolver soluções inovadoras e evoluir as já existentes inserindo mais inovação consistem nas principais atividades. Os softwares desenvolvidos são essenciais na tomada de decisão na maior empresa do setor elétrico brasileiro, o que demonstra a qualidade, complexidade e importância das soluções desenvolvidas.

A Smartiks possui capacidade reconhecida no desenvolvimento de soluções inteligentes para empresas de geração e distribuição de energia elétrica. Entre os principais projetos de P&D mantidos e evoluídos pela Smartiks, três merecem uma atenção especial:

- Tratamento inteligente de alarmes on-line de subestações e centros de controle;

- Determinação inteligente de ações para recomposição da rede de transmissão de energia;
- Geração automática de procedimentos de liberação e normalização de equipamentos de uma subestação de energia elétrica.

A experiência com P&D da Smartiks vai desde a concepção até a comercialização dos seus resultados. A Smartiks será a primeira empresa a receber uma licença de comercialização de um resultado de P&D na Chesf (o licenciamento foi aprovado pela diretoria da Chesf em setembro/2013).

O grande diferencial da Smartiks está em sua equipe e na sua capacidade de transformar em produto os resultados de projetos de P&D. A parceria com a academia faz com que a inovação seja parte de sua missão. Os sócios da Smartiks têm mais de 10 anos de experiência com P&D. Todas essas características fazem da Smartiks uma empresa com capacidade de executar a pesquisa e implantá-la no mercado.

1.3 ESTRUTURA DO TRABALHO

Este relatório está dividido em quatro capítulos. O Capítulo 1 compreende a presente introdução, em que são apresentados os objetivos deste trabalho e uma breve descrição da empresa em que o estágio supervisionado ocorreu.

No capítulo 2 é apresentada uma breve fundamentação teórica sobre potência elétrica em circuitos em regime senoidal, microcontroladores e uma introdução sobre o conceito de internet das coisas e o protocolo de comunicação MQTT.

No capítulo 3 são descritas as atividades desenvolvidas pelo estagiário na empresa durante o período de estágio.

Por fim, o capítulo 4 descreve as considerações finais e perspectivas futuras no que concerne ao trabalho desenvolvido no estágio.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 POTÊNCIA ELÉTRICA EM REGIME PERMANENTE SENOIDAL

Na maioria dos casos, o fornecimento de energia elétrica para realização de trabalho é feito de forma alternada. Ou seja, este fornecimento é dado na forma de tensões e correntes variando no tempo de forma periódica.

Nesta seção são apresentados alguns conceitos essenciais no que diz respeito à potência em circuitos em regime permanente senoidal.

- **Potência Instantânea**

A potência instantânea em algum componente/ponto de um circuito, em um determinado instante de tempo t , é encontrada efetuando-se o produto entre a tensão v e a corrente i em neste instante de tempo t .

Desta forma, define-se a potência instantânea $p(t)$ como sendo:

$$p(t) = v(t)i(t) \quad (1)$$

Caso a corrente elétrica i esteja no sentido da elevação de tensão, a equação acima deve apresentar um sinal negativo. A unidade da potência é expressa em *watts* (W).

- **Potência Média ou Ativa**

Define-se a potência média como sendo “a média das potências instantâneas durante um período ou, em forma de equação,

$$P = \frac{1}{T} \int_0^T p(t) dt \quad (2)$$

Onde T é o período da função” (NILSSON, RIEDEL, 2009)

Para o caso específico em que a tensão e a corrente são puramente senoidais, o valor da potência ativa também pode ser calculado pela equação a seguir:

$$P = V I \cos\phi \quad (3)$$

Onde V e I são os valores eficazes de $v(t)$ e $i(t)$, e o ângulo ϕ representa o ângulo de defasagem entre a tensão e a corrente.

A potência média é também chamada de potência real ou ativa. Ela corresponde à energia efetivamente consumida pela carga. No sistema Internacional de Unidades, é medida em *watts* (W). Em indutores e capacitores, tem-se $P = 0$ (GUERRA, 2011).

- **Potência Reativa**

Define-se *potência reativa* (Q) como sendo a componente da potência instantânea associada aos elementos armazenadores de energia (indutores e capacitores).

A potência reativa é expressa em volt-ampéres reativos (var) e pode ser calculada pela equação a seguir:

$$Q = V I \operatorname{sen}\phi \quad (3)$$

Nas redes de energia elétrica, a existência de elevados valores de potência reativa circulando nas linhas constitui um fator indesejável, pois ocorrem perdas ôhmicas nos condutores e quedas de tensão nos terminais da carga. Além disso, um consumidor que requer excesso de potência reativa do sistema é penalizado com um acréscimo na sua conta de consumo de energia (GUERRA, 2011).

- **Potência Aparente**

Define-se *potência aparente* como sendo o produto dos valores eficazes da tensão e da corrente, ou seja:

$$S = VI \quad (4)$$

A potência aparente é medida em volts-ampéres (VA). É a potência que se utiliza na especificação de equipamentos elétricos, como geradores, transformadores, etc. É mais usual utilizar seus múltiplos (kVA, MVA) (GUERRA, 2011).

- **Fator de Potência**

Define-se *fator de potência* de uma instalação como a relação entre a potência ativa consumida e a potência aparente, ou seja:

$$FP = \frac{POTÊNCIA\ ATIVA}{POTÊNCIA\ APARENTE} \quad (5)$$

No caso senoidal, esta definição resume-se a:

$$FP = \cos(\phi) \quad (6)$$

Onde ϕ é o ângulo de defasagem entre a tensão e a corrente.

- **Triângulo de Potências**

É usual relacionar S, P, Q e ϕ através do triângulo de potências, conforme mostrado na figura abaixo:

Fonte: **NILSSON, RIEDEL, 2009**

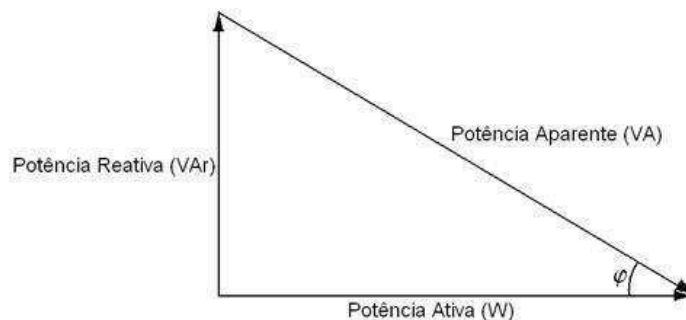


FIGURA 2: TRIÂNGULO DE POTÊNCIAS

A representação gráfica das potências permite que diversos problemas de solução analítica laboriosa possam ser mais facilmente resolvidos através de recursos da geometria euclidiana, como o teorema de Pitágora, lei dos senos, lei dos cossenos, etc (**GUERRA, 2011**).

2.2 MICRONCONTROLADORES

Os microcontroladores são circuitos eletrônicos que reúnem num único chip: uma unidade de processamento, memórias e um conjunto de periféricos de entrada e saída.

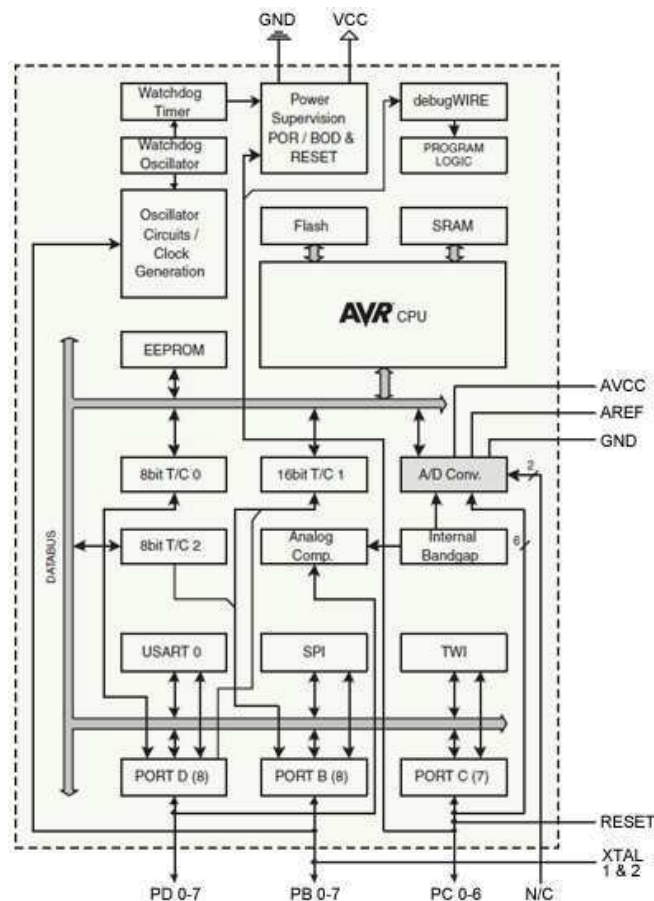
Isto facilita o desenvolvimento de produtos eletrônicos, pois, com um único componente, pode-se desenvolver aplicações bastante complexas. Além disso, o projeto das placas é simplificado, visto que a quantidade de componentes e ligações fica reduzida.

De forma geral, os microcontroladores possuem memória volátil de alta velocidade de acesso (memória RAM, os valores são perdidos depois que a fonte de alimentação é retirada) e memória não volátil para armazenamento de programas

(memória *Flash* ou EEPROM) no microcontroladores mais atuais, onde os dados continuam gravados mesmo depois da retirada da fonte (ALMEIDA,2016).

Na figura abaixo, é apresentado o diagrama de blocos do microcontrolador ATmega328 da empresa *Atmel*:

FIGURA 3: DIAGRAMA DE BLOCOS DO ATMEGA328



Fonte: *Datasheet* do ATmega328

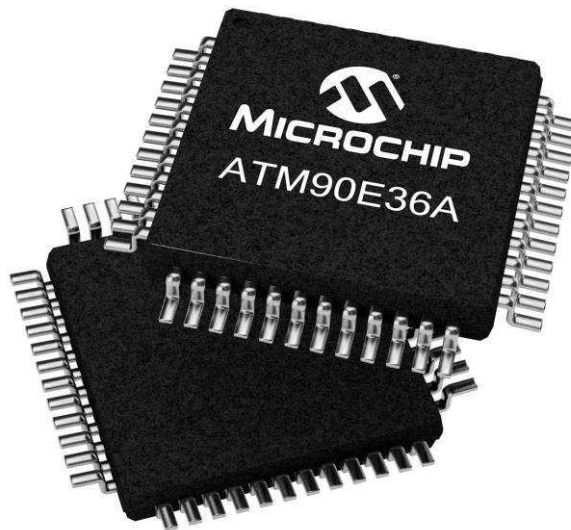
Os microcontroladores da família Atmel 8-bit são baseados na arquitetura AVR, um modelo de processador RISC. Essa arquitetura possui 8 bits e barramentos separados de dados e programas (arquitetura Harvard). Esta arquitetura foi desenvolvida em 1996, sendo um dos primeiros microcontroladores a fazerem uso de memória flash para armazenar o código do programa. Isso permitiu que os chips fossem reprogramados de modo simples, facilitando o processo de desenvolvimento do produto (ALMEIDA,2016).

2.3 *POWER MANAGEMENT INTEGRATED CIRCUIT – PMIC*

Os PMICs são circuitos integrados cuja funcionalidade é voltada a gerência de consumo de potência em um determinado sistema. De forma geral, estes dispositivos possuem as seguintes características:

- Cálculos internos de grandezas como: potências, valores eficazes, frequência, energia, etc.;
- Erro abaixo de 1% para os cálculos internos
- Vários canais de ADC
- Baixo consumo de potência (corrente na faixa de mA para uma alimentação de 5 V ou 3.3 V contínua)
- Interface de comunicação digital (variando de acordo com o chip, podendo ser SPI, I2C, UART, etc.).
- Tamanho pequeno, feitos em tecnologia SMD

FIGURA 4: PMIC AM90336A



Fonte: Site Microchip

2.4 INTERNET DAS COISAS

A Internet é um sistema global de redes de computadores que usa a pilha de protocolos (TCP/IP) para conectar dispositivos ao redor do mundo.

A internet surgiu a partir de pesquisas militares no auge da Guerra Fria(1969) e inicialmente foi utilizada para a troca de mensagens entre universidades e centros de pesquisas com intuito de compartilhar informações valiosas e descentralizar as mesmas.

Com o passar do tempo, a Internet se tornou bastante popular, deixando assim de ser usada apenas para conectar computadores entre si, mas também para conectar pessoas por meio de redes sociais. Hoje em dia, milhões de pessoas criam perfis pessoais e profissionais e se conectam umas as outras por meio da rede de computadores.

No cenário atual, uma nova fase da internet se encontra em ascensão. A esta nova etapa da Internet dá-se o nome de **Internet das Coisas**. Nesta nova tendência, a internet está sendo usada para conectar dispositivos eletrônicos utilizados no dia a dia(coisas) como aparelhos eletrodomésticos, máquinas industriais, meios de transporte, entre muitos outros. A internet das coisas (*IoT*) permite que objetos de diferentes naturezas sejam monitorados, controlados e que troquem informações entre si. Isto possibilita uma maior integração entre o mundo físico e o mundo dos sistemas computacionais, o que resulta em uma melhora da eficiência, precisão e benefícios econômicos por parte dos usuários.

Diante disso, é necessário que haja um padrão de comunicação entre os dispositivos conectados de forma que eles consigam conversar entre si de forma interoperável. Uma vez que, estes dispositivos, em sua maioria, possuem recursos de memória, processamento e energia bastante limitados, protocolos já consagrados como o http não representam uma ótima alternativa para a internet das coisas. Em consequência disso, estão surgindo alguns protocolos desenhados exclusivamente para estas aplicações e cujo levam em conta as limitações acima citadas.

Na subseção seguinte é apresentada uma breve descrição do protocolo de comunicação MQTT. Este protocolo vem ganhando bastante atenção no cenário da Internet das Coisas por ser um protocolo simples e leve, adequado para aplicações em *IoT*.

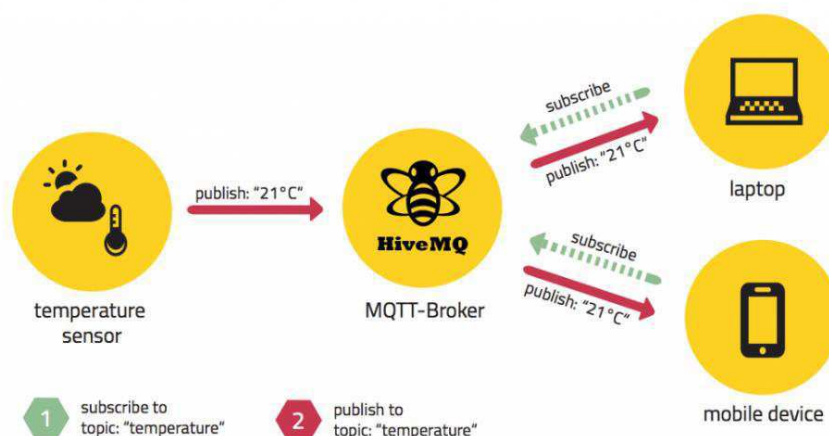
2.4.1 PROTOCOLO DE COMUNICAÇÃO MQTT

O MQTT é um protocolo de comunicação da camada de aplicação do modelo Ineternet (funciona no topo da pilha TCP/IP) desenvolvido em 1999 pela empresa IBM.

Este protocolo é baseado na arquitetura *publish/subscribe* e foi projetado para troca de mensagens entre dispositivos que possuem recursos disponíveis limitados (memória, bateria, capacidade de processamento, etc).

Por ser simples, leve e ao mesmo tempo robusto o suficiente para garantir que mensagens importantes cheguem ao seu destino a todo momento, o MQTT é ideal para aplicações que envolvam comunicação *machine to machine* (M2M) , Internet das Coisas e aplicações móveis.

FIGURA 5: MQTT - PADRÃO *PUBLISH/SUBSCRIBE*



Fonte: Site *HiveMQ*

Na figura acima é ilustrada a arquitetura do protocolo de comunicação MQTT. Neste padrão, quando um dispositivo da rede deseja receber uma determinada informação ele faz uma requisição para um outro elemento da rede capaz de gerir o fluxo de mensagens entre os clientes. No protocolo MQTT este elemento é denominado *broker* e é responsável por intermediar a comunicação entre os elementos da rede.

A identificação do conteúdo das mensagens no MQTT é feita através de estruturas denominadas tópicos. Tópicos são uma cadeia de caracteres que representam o “assunto” das mensagens que estão sendo transmitidas naquele canal. Na figura acima, por exemplo, o dispositivo medidor de temperatura está enviando mensagens para o servidor no tópico

“*temperature*”. Os outros dois dispositivos da rede assinaram (*subscribe*) o respectivo tópico e passaram a receber os valores de temperatura medidos pelo sensor. Desta forma, quando o *broker* receber uma mensagem com uma nova medição de temperatura esta mensagem será imediatamente encaminhada para todos os outros clientes que assinaram o respectivo tópico.

3 ATIVIDADES DESENVOLVIDAS

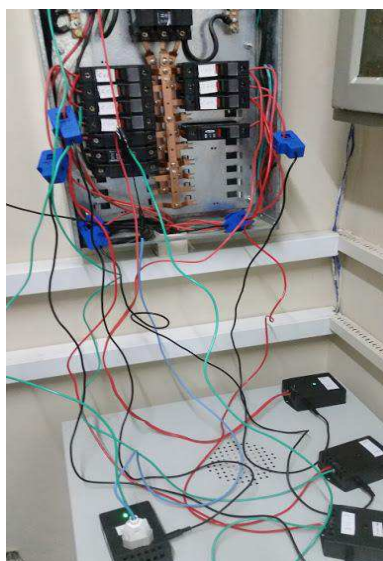
3.1 ACOMPANHAMENTO NA INSTALAÇÃO DOS MEDIDORES

A primeira atividade desenvolvida durante o período de estágio supervisionado foi o acompanhamento na instalação de alguns medidores de energia elétrica inteligentes desenvolvidos pela empresa.

Durante esta etapa, foram realizadas algumas visitas em instalações prediais com o intuito de realizar a instalação dos medidores inteligentes monofásicos não invasivos desenvolvidos pela empresa. O objetivo destas instalações é verificar se os medidores estão funcionando de forma adequada e coletar dados da rede elétrica em questão para serem posteriormente analisados pela equipe de engenharia de *software* da empresa.

A seguir é apresentada uma imagem de alguns medidores monofásicos que foram instalados em uma das visitas.

FIGURA 6: MEDIDORES MONOFÁSICOS INTELIGENTES INSTALADOS



Fonte: Elaborada pelo autor

Uma vez instalados, os medidores coletam valores de corrente e de tensão da rede elétrica e os envia para um servidor da empresa por meio da internet. A seguir, é apresentada a imagem de um dos medidores corretamente instalado e em pleno funcionamento. A luz verde do led RGB de status indica que a conexão com o servidor foi bem sucedida e que o medidor está a enviar dados.

FIGURA 7: MEDIDOR INTELIGENTE ENVIANDO DADOS PARA O SERVIDOR



Fonte: Elaborada pelo próprio autor

3.2 MANUAL DE INSTALAÇÃO MEDIDORES INTELIGENTES

Uma outra atividade desenvolvida durante o estágio supervisionado foi a escrita de um manual de instalação para os medidores inteligentes.

O manual de instalação escrito teve como objetivo descrever os medidores desenvolvidos pela empresa e os passos a serem seguidos para uma correta instalação dos mesmos.

O documento contempla os medidores inteligentes trifásicos e monofásicos não invasivos desenvolvidos pela empresa e , além da instalação elétrica, também visa instruir o consumidor a como visualizar informações da rede elétrica de sua residência/empresa por meio de um aplicativo móvel também desenvolvido pela empresa. Estas informações incluem: potência ativa, potência reativa, valores RMS, fator de potência, dentre outras grandezas. Dentre os tópicos abordados no manual estão:

- Instruções de Segurança;
- Informações sobre o manual;
- Descrição Física e Funcional dos medidores desenvolvidos pela empresa;
- Instruções de Instalação Elétrica dos Medidores;
- Instruções de conectar os medidores ao WiFi por meio do uso de uma aplicação móvel desenvolvida pela empresa;
- *Frequently Asked Questions* (FAQ);

3.3 UPLOAD DE *FIRMWARE* EM MICROCONTROLADORES

Outra atividade desempenhada durante o estágio supervisionado foi a passagem do *firmware* final do produto para os microcontroladores utilizados nos medidores da empresa.

O microcontrolador utilizado nesta atividade foi o Atmega328 da empresa Atmel, na figura abaixo é apresentada uma imagem deste microcontrolador.

FIGURA 8: MICROCONTROLADOR ATMEGA328

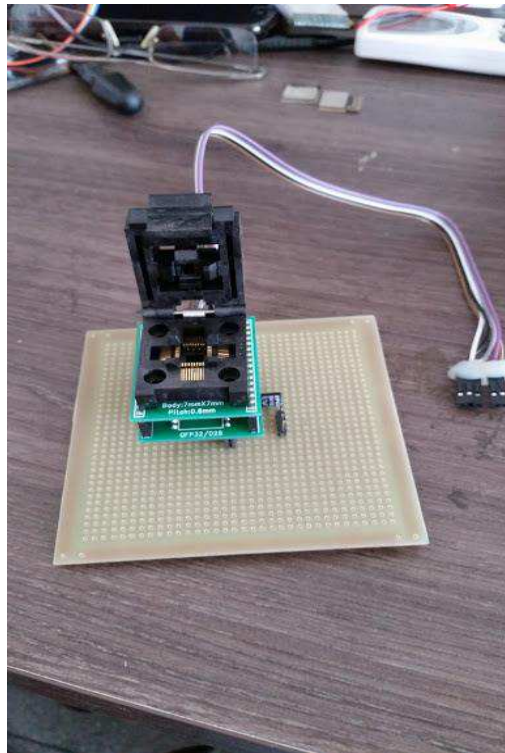


Fonte: Site Instituto Digital

Uma vez que os microcontroladores utilizados eram novos e não possuíam nenhum programa escrito em sua memória *flash*, foi necessário efetuar a gravação do *bootloader* nestes microcontroladores. O *bootloader* é um programa especial localizado normalmente no início da memória *flash* do microcontrolador e é responsável por facilitar a gravação de outros programas que o usuário deseje rodar no microcontrolador.

Devido à incompatibilidade dos pinos deste microcontrolador em específico com as protoboards convencionais, fez-se necessário o uso de um adaptador tqfp32. Este dispositivo possui um encaixe específico para a pinagem do microcontrolador utilizado e possibilita que o mesmo seja programado pelo computador de forma mais prática. A seguir é mostrada uma imagem do dispositivo utilizado na atividade.

FIGURA 9: ADAPTADOR SOQUETE TQFP32



Fonte: Elaborada pelo autor

Nesta etapa do estágio foi realizada a programação dos microcontroladores a serem utilizados nos circuitos dos medidores inteligentes da empresa. Uma vez programados, os microcontroladores foram encaminhados para o processo de confecção de placa de circuito impresso.

3.4 DESENVOLVIMENTO DE UM SISTEMA EMBARCADO PARA CONTROLE DE TEMPERATURA VIA INTERNET

Esta atividade teve como objetivo solucionar um problema no ambiente de trabalho da equipe de *hardware* da empresa.

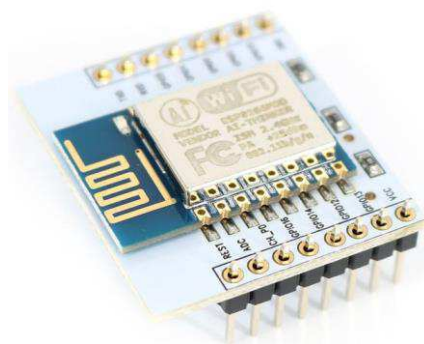
Tendo em vista que o ar condicionado instalado na sala leva muito tempo para resfriar o ambiente, decidiu-se desenvolver, como parte do estágio supervisionado, um sistema embarcado que possibilita-se às pessoas que utilizam o ambiente, controlar o ar condicionado e monitorar a temperatura atual na sala de forma remota e em tempo real por meio de uma página na internet.

Isto possibilitou aos usuários da sala saberem qual a temperatura da sala a qualquer momento e , caso desejem, ligarem o ar condicionado da sala e setar a sua temperatura antes de chegar à sala. Desta forma, quando o funcionário chegar em seu ambiente de trabalho a sala já estará devidamente climatizada.

O microcontrolador utilizado neste projeto foi o ESP8266. O ESP8266 é um microcontrolador de baixo custo, com suporte WiFi e possui as seguintes especificações:

- Wireless padrão 802.11 b/g/n;
- Antena embutida;
- Modos de operação: STA/AP/STA+AP;
- Segurança WEP, WPA, TKIP, AES;
- Protocolo TCP/IP embutido;
- Portas GPIO: 11;
- Tensão de operação: 3,3V;
- Taxa de transferência: 110-460800bps;
- Conversor analógico digital (ADC);
- Distância entre pinos: 2mm;
- Dimensões: 24 x 16 x 3,2 mm ;

FIGURA 10: ESP8266 ENCAPSULADO COMO ESP-12



Fonte: Site *Deal Extreme*

Um LED emissor infravermelho foi utilizado para enviar os sinais responsáveis por se comunicar com o ar condicionado. O led foi conectado à porta digital 12 do microcontrolador e possui as seguintes especificações:

- Tensão de operação: 1.2 à 1.4V
- Diâmetro: 5mm
- Comprimento de onda: 940nm
- Terminais: 2

FIGURA 11: LED EMISSOR INFRAVERMELHO



Fonte: Site *filipeplop*

A aquisição da temperatura do ambiente foi feita por meio do uso do sensor de temperatura LM35. Este sensor foi conectado a entrada analógica do ESP8266 e possui as seguintes características:

FIGURA 12: SENSOR DE TEMPERATURA LM35



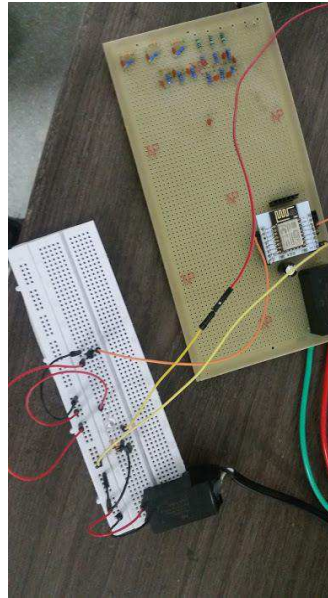
Fonte: Site *filipeflop*

- Faixa de temperatura: -0°C a 100°C
- Precisão: 0.5 °C
- Calibrado em graus Celsius
- Tensão de operação: 4 a 30V
- Consumo de corrente: <60 μ A

O microcontrolador foi programado por meio da IDE Arduino e o protocolo de comunicação MQTT foi utilizado para a troca de mensagens pela internet. Isto

possibilitou ao sistema enviar os valores de temperatura da sala para um servidor e receber os comandos necessários para o controle do ar condicionado. A seguir, é mostrado o protótipo do projeto montado em uma placa de prototipagem:

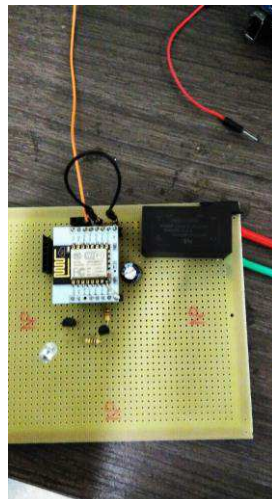
FIGURA 13: PROTÓTIPO DO PROJETO



Fonte: Elaborada pelo autor

Uma vez que o protótipo foi testado e validado, o mesmo foi montado e soldado em uma placa de circuito padrão, conforme mostra a imagem a seguir:

FIGURA 14: VERSÃO FINAL DO PROJETO

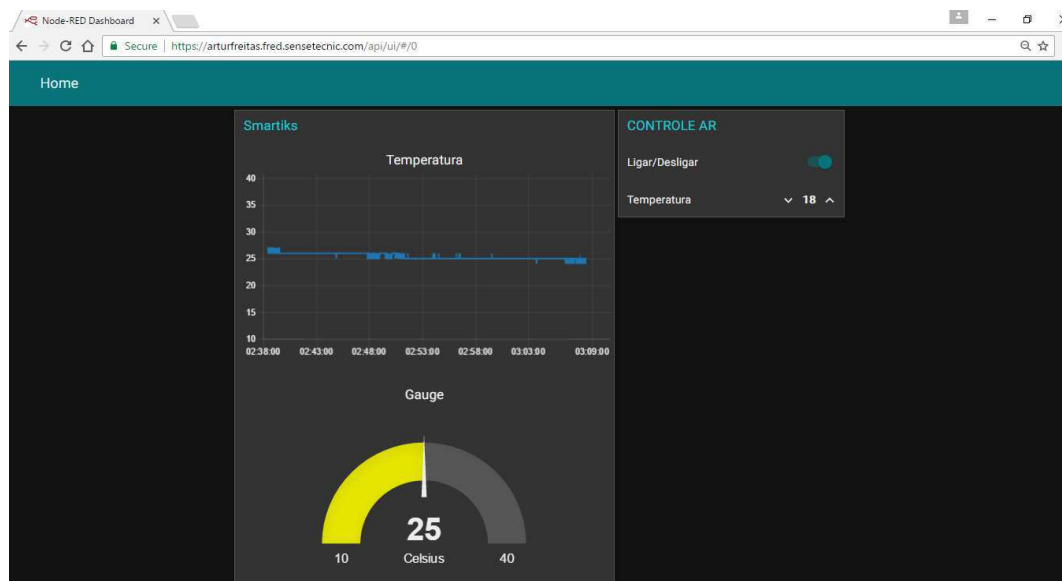


Fonte: Elaborada pelo autor

Para proporcionar que o usuário pudesse interagir com o sistema foi criada uma interface web, utilizando a ferramenta *Node Red*. O *Node Red* é uma ferramenta

desenvolvida pela IBM e facilita o desenvolvimento de projetos envolvendo Internet das Coisas. A seguir é mostrada a interface criada:

FIGURA 15: INTERFACE WEB



Fonte: Elaborada pelo próprio autor

Por meio desta interface, é possível monitorar em tempo real a temperatura da sala por meio de um gráfico e controlar o ar condicionado remotamente através de botões presentes na página.

4 CONCLUSÃO

4.1 CONSIDERAÇÕES FINAIS

O estágio supervisionado na empresa Smartiks, além de necessário para a formação acadêmica, foi de suma importância para a formação profissional do aluno, pois foi uma oportunidade para colocar em prática os conhecimentos adquiridos durante os anos de graduação e entrar em contato com o mercado de trabalho.

As atividades desenvolvidas pelo aluno envolveram diversos âmbitos da engenharia elétrica, dentre eles pode-se citar: Instalações elétricas, programação de sistemas embarcados, protocolos de comunicação, instrumentação eletrônica e redes de computadores.

4.2 PERSPECTIVAS FUTURAS

Dentre as perspectivas futuras do trabalho desenvolvido durante o estágio supervisionado pode-se citar a edição do manual de instalação criado caso ocorra a adição de novas funcionalidades nos medidores abordados, uma vez que o projeto dos medidores se encontra em fase final de desenvolvimento e está sujeito a sofrer alterações em seu *design* e funcionamento.

Além disso, o sistema de controle de ar condicionado pela internet desenvolvido durante o estágio pode ser aplicado à qualquer ambiente que possua ar condicionado e possui potencial para ser utilizado em outros sistemas de automação residencial e predial.

REFERÊNCIAS

Espressif Systems. Datasheet: ESPRESSIF SMART CONNECTIVITY PLATFORM: ESP8266. Publicação eletrônica. 2013.

GUERRA, F.C.F. *Notas de aula: Circuitos Elétricos II*. Campina Grande, 2011

NILSSON, James W. RIEDEL, Susan A. *Circuitos Elétricos*. 8 Ed. São Paulo: Pearson Prentice Hall, 2009.

Atmel. Datasheet: ATMEL 8-BIT MICROCONTROLLER WITH 4/8/16/32KBYTES IN-SYSTEM PROGRAMMABLE FLASH. Publicação eletrônica. 2015.

Documentos do projeto da SMARTIKS TECNOLOGIA DA INFORMAÇÃO LTDA – ME

EMBARCADOS. MQTT – PROTOCOLOS PARA IOT. Disponível em:
<https://www.embarcados.com.br/mqtt-protocolos-para-iot/>. Acesso em Abril de 2017.

ALMEIDA, Rodrigo Maximiano Antunes de. *Programação de Sistemas Embarcados: Desenvolvendo software para microcontroladores em linguagem C/* Rodrigo Maximiano Antunes de Almeida, Carlos Henrique Valério Moraes, Tatyana de Faria Piola Seraphim. Elsevier, 2016.

APÊNDICE A – CÓDIGO FONTE DO PROJETO

DESENVOLVIDO NA ATIVIDADE 4

```
//projeto_ar.ino
#include <IRremoteESP8266.h>
#include <IRremoteInt.h>
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

#define IR_EMITER    12

//setando o nome da rede WiFi e a senha
const char* ssid = "set-wireless";
const char* password = "smart02-";

//declarando variáveis utilizadas para leitura do ADC e cálculo da temperatura
int adc_value;
float temp_celsius;
char temp[5];

//Instanciando um objeto do tipo IRsend e definindo o pino a ser utilizado
IRsend irsend(IR_EMITER);

//Definindo a frequência de modulação (normalmente é 38Khz)
int khz=38; //NB Change this default value as necessary to the correct modulation
frequency

// Setando o endereço IP do Broker utilizado (broker.hivemq.com)
IPAddress broker(35, 157, 21, 220); //Server IP
```

//Função callback que é executada quando alguma mensagem chega proveniente do servidor mqtt

```
void callback(char* topic, byte* payload, unsigned int length) {
```

```
    String inputString = "";
```

```
    //Processando a mensagem e a armazenando na variável inputString
```

```
    for (int i=0;i<length;i++) {
```

```
        inputString += (char)payload[i];
```

```
    }
```

//Comparando a inputString e enviando o respectivo comando para o emissor infravermelho

```
    // Ligar
```

```
    if(inputString == "1"){
```

```
        unsigned int  rawData[99] = {9100,4350, 650,1550, 700,1500, 650,500,
650,500, 650,500, 600,550, 600,600, 500,1700, 500,650, 500,1750, 500,1700, 500,650,
500,650, 500,650, 500,650, 450,700, 450,700, 450,700, 450,1800, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,1750, 450,700, 450,700, 450,700, 450,1800,
450,650, 500,700, 450,700, 350,650, 450,700, 450,750, 400,700, 450,700, 450,700,
450,700, 450,700, 450,650, 400,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450}; // UNKNOWN B8E1DD93
```

```
        Serial.println("Entrou 1");
```

```
        irsend.sendRaw(rawData,99,38);
```

```
        delay(10);
```

```
    }
```

```
    //Desligar
```

```
    else if(inputString == "0"){
```

```

    unsigned int  rawData[99] = {9050,4400, 650,1550, 700,1550, 600,550,
600,550, 600,550, 600,550, 600,550, 550,1700, 500,650, 500,1700, 500,1700, 500,650,
500,650, 500,650, 500,650, 500,650, 450,700, 450,700, 450,1750, 500,650, 500,650,
500,650, 500,650, 500,650, 500,650, 500,1750, 450,700, 450,700, 450,700, 450,1750,
500,650, 500,650, 500,650, 500,650, 500,650, 500,650, 500,650, 500,650, 500,650,
500,650, 500,650, 500,650, 500,650, 500,650, 500,650, 500,650, 500,650,
500}; // UNKNOWN 16DABDB5
    irsend.sendRaw(rawData,99,38);
    delay(10);
}
//Setar a temperatura do ar para 18°C
else if(inputString == "18"){
    unsigned int  rawData[99] = {9150,4300, 750,1450, 700,1550, 600,550,
600,550, 600,550, 550,650, 500,650, 450,1750, 500,650, 500,1700, 450,1800, 450,700,
450,700, 450,700, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750,
400,750, 400,750, 400,750, 400,750, 400,1800, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 400,750, 400,750, 400,750, 400,750, 400,700, 350,750,
400}; // UNKNOWN 8E180C61
    irsend.sendRaw(rawData,99,38);
    delay(10);
}
//Setar a temperatura do ar para 19°C
else if(inputString == "19"){
    unsigned int  rawData[101] = {200,250, 9150,4300, 800,1450, 700,1500,
650,500, 650,550, 550,600, 550,600, 500,650, 500,1750, 500,650, 450,1750, 500,1750,
400,700, 450,700, 450,700, 450,700, 450,700, 400,750, 400,750, 400,750, 400,750,
400,750, 350,750, 400,750, 400,750, 400,750, 400,1800, 450,700, 450,700, 450,1800,
400,750, 400,750, 400,750, 400,700, 450,700, 450,700, 450,700, 450,700, 450,700,
400,750, 400,750, 400,750, 400,750, 400,700, 400,750, 400,750, 400,750, 400,750,
400,700, 450}; // UNKNOWN C5253CD7
    irsend.sendRaw(rawData,99,38);
    delay(10);
}

```

```

//Setar a temperatura do ar para 20°C
else if(inputString == "20"){
    unsigned int  rawData[99] = {9100,4400, 650,1550, 650,1550, 650,500,
650,550, 600,550, 600,550, 550,600, 550,1650, 550,600, 550,1650, 500,1750, 500,650,
500,650, 500,650, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,1800, 450,700, 450,700, 450,700, 450,1750,
450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,750,
400,750, 400,750, 400,750, 400,750, 450,700, 400,750, 400,750, 400,750, 400,750,
400}; // UNKNOWN E2D339D3

    irsend.sendRaw(rawData,99,38);
    delay(10);
}

//Setar a temperatura do ar para 21°C
else if(inputString == "21"){
    unsigned int  rawData[99] = {9200,4250, 750,1500, 700,1500, 650,500,
650,550, 550,600, 550,600, 500,650, 500,1750, 450,650, 500,1750, 450,1750, 450,700,
450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,650, 350,1850, 400,750, 400,750, 400,1800, 400,1850,
400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750,
400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750,
400}; // UNKNOWN 849FDE16

    irsend.sendRaw(rawData,99,38);
    delay(10);
}

//Setar a temperatura do ar para 22°C
else if(inputString == "22"){
    unsigned int  rawData[99] = {9200,4250, 750,1500, 700,1500, 700,450,
650,500, 600,550, 600,550, 550,650, 500,1700, 500,650, 500,1750, 450,1750, 450,700,
450,700, 450,700, 450,750, 400,750, 400,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,750, 400,700, 450,1800, 400,750, 400,750, 400,750, 400,750,
400,1800, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,750, 400,750, 400,700, 450,700, 450,700, 450,700, 450,700, 450,700,
400}; // UNKNOWN 527E58D7

```

```

    irsend.sendRaw(rawData,99,38);
    delay(10);
}
    //Setar a temperatura do ar para 23°C
    else if(inputString == "23"){
        unsigned int  rawData[99] = {9100,4300, 750,1500, 650,1550, 650,500,
650,550, 600,550, 550,600, 500,650, 500,1750, 500,650, 500,1700, 500,1750, 500,650,
450,700, 450,700, 450,650, 450,750, 400,750, 400,750, 400,750, 400,700, 400,750,
400,750, 400,750, 400,750, 400,700, 400,1850, 400,700, 450,700, 450,1800, 400,750,
400,1800, 450,700, 450,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750,
400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750, 400,750,
400}; // UNKNOWN 80BB49D9
        irsend.sendRaw(rawData,99,38);
        delay(10);
    }
        //Setar a temperatura do ar para 24°C
        else if(inputString == "24"){
            unsigned int  rawData[99] = {9050,4350, 650,1600, 650,1550, 650,500,
650,500, 650,500, 650,500, 600,550, 550,1700, 500,650, 500,1700, 550,1700, 500,650,
500,650, 500,650, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,1800, 450,700, 450,700, 450,700, 450,1750,
450,1800, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450}; // UNKNOWN A065EFDF
            irsend.sendRaw(rawData,99,38);
            delay(10);
        }

            //Setar a temperatura do ar para 25°C
            else if(inputString == "25"){
                unsigned int  rawData[99] = {9050,4400, 650,1550, 650,1600, 600,550,
600,550, 600,550, 600,550, 550,1700, 550,600, 500,1700, 550,1700, 500,650,
500,650, 500,650, 500,650, 500,650, 500,650, 500,700, 400,750, 400,750, 400,750,
400,750, 400,750, 450,700, 400,750, 450,1750, 450,700, 450,700, 450,1800, 400,1800,

```

```

450,1750, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700, 450,700,
450,700, 450,700, 450,700, 450,700, 450,650, 450,700, 450,700, 450,700, 450,700,
450}; // UNKNOWN 6795432F
    irsend.sendRaw(rawData,99,38);
    delay(10);
}

}

//Instanciando objetos responsáveis pela conexão wifi e conexão MQTT
WiFiClient wifiClient;

PubSubClient client(wifiClient);

//Função responsável por reconectar com o servidor mqtt caso a conexão não tenha
sido estabelecida
void reconnect() {

    while (!client.connected()) {

        //Se conecta com o servidor mqtt
        if (client.connect("espClient")) {

            // Assina o tópico smartiks/ar
            client.subscribe("smartiks/ar");
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            delay(5000);
        }
    }
}
}

```

```
void setup() {

    irsend.begin(); // Inicia o processo de envio dos sinais infravermelhos
    delay(30);

    pinMode(A0, INPUT); // Define o pino A0 como entrada

    client.setServer(broker, 1883); // seta o servidor e a porta
    client.setCallback(callback); //seta a função callback

    WiFi.begin(ssid, password); // Inicia a conexão com wifi

    while (WiFi.status() != WL_CONNECTED) { //Aguarda a conexão ser
estabelecida
        delay(500);
    }
}

void loop() {

    if (!client.connected()) {
        reconnect();
    }

    adc_value = analogRead(A0); // Efetua a leitura do sensor de temperatura
    temp_celsius = adc_value*0.09765625; // Converte o valor de tensão lido para
Celsius
    sprintf(temp, "%f", temp_celsius); //Escreve este valor em um array de caracteres
```

```
    client.publish("smartiks/temp", temp); //Publica o valor da temperatura atual no
tópico smartiks/temp
    delay(100);
    client.loop(); //Verifica se a conexão com o servidor mqtt está ativa

}
```