



UNIVERSIDADE FEDERAL DA PARAÍBA - UFPB
CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS - CEAR
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA



Gustavo Pessoa de Pontes

Relatório de Estágio

Estágio Supervisionado na Empresa Fácil Soluções Tecnológicas

João Pessoa
2018

Gustavo Pessoa de Pontes

Estágio Supervisionado na Empresa Fácil Soluções Tecnológicas

Relatório de Estágio Integrado submetido ao Departamento de Engenharia Elétrica como um dos pré-requisitos para a obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Prof. Dr. Fabrício Braga Soares de Carvalho

Identificação

EMPRESA:

Nome: Fácil Soluções Tecnológicas

Endereço: Av. Paraíba, 45

Bairro: Estados

Cidade/Estado: João Pessoa/Paraíba

Telefone: +55 083 3513-9762

CEP: 58030-430

ESTÁGIO:

Área da Instituição: Setor de Pesquisa e Desenvolvimento – Fácil Soluções Tecnológicas

Data de Início: 5/05/2018

Carga Horária Semanal: 30

Carga Horária Total: 180 H

Supervisor de Estágio: Rodrigo de Dantas Freitas

Orientador do Estágio: Prof. Dr. Fabrício Braga Soares de Carvalho

Gustavo Pessoa de Pontes

Estágio Supervisionado na Empresa Fácil Soluções Tecnológicas

Relatório de Estágio Integrado submetido ao Departamento de Engenharia Elétrica como parte dos requisitos necessários à obtenção do grau de Bacharel em Engenharia Elétrica.

Data de Aprovação: ____/____/____

Nota: ____

Prof. Dr. Fabício Braga Soares de Carvalho (Orientador)
Universidade Federal da Paraíba

Prof. Dr. Jose Mauricio Ramos de Souza Neto (Avaliador)
Universidade Federal da Paraíba

Prof. MSc. Rafael de Sousa Marinho (Avaliador)
Universidade Federal da Paraíba

Resumo

Este relatório apresenta as principais atividades executadas durante o estágio no setor de Pesquisa e Desenvolvimento da empresa Fácil Soluções Tecnológicas. Realizaram-se estudos e desenvolvimento de circuitos eletrônicos para diferentes aplicações de comunicações sem fio, dentre as quais destacam-se: controle de acesso a setores da empresa; circuitos para comunicação entre sistemas *web*; uma balança eletrônica; e um circuito para alertar os horários de ponto para os funcionários.

Palavras-chave: Internet das Coisas; ESP8266; Comunicação Sem Fio.

Abstract

This report presents the main activities executed during the internship at the Research and Development Department of Fácil Soluções Tecnológicas company. In this company were executed studies and development of electronics circuits for different wireless communications applications, among which the following stand out: control of access to sectors of the company; circuits for communication between web systems; an electronic weighing-machine; and a circuit to alert employees of check in/out on time attendance systems.

Keywords: Internet of Things; ESP8266; Wireless Communication.

Agradecimentos

Em primeiro lugar agradeço a Deus por ter me iluminado por toda a minha trajetória até este momento, sem Ele nada disto seria possível.

Aos meus pais Ana Lucia Pontes e Odivio Pontes, a minha irmã Germana Pontes e aos meus avós Odilon Pontes e Maria Ana Pontes que foram a minha base durante todo este tempo, me incentivaram e sempre acreditaram em mim sob qualquer circunstância ao qual passei. Minha família é simplesmente a coisa mais importante que tenho na vida.

Ao meu orientador Fabrício Braga, pelo apoio, paciência e pela ajuda durante o estágio, como também ao longo do curso.

A todos os professores que contribuíram para a minha formação e que me guiaram pelo caminho do aprendizado.

As pessoas da empresa Fácil Soluções Tecnológicas que foram receptivos durante o estágio.

Aos meus amigos de longa data Michael Cavalcante e Rennan Martins que me ajudaram não só com problemas relacionados com a universidade, como também a crescer pessoalmente. Agradeço a eles os momentos de companheirismo que me proporcionaram momentos dos quais nunca vou esquecer.

Aos meu colegas de curso, em especial Jean Cardoso e Douglas Medeiros que me ajudaram a passar pelos momentos mais difíceis do curso, como também me incentivaram a nunca desanimar quando tudo dava errado.

Por fim, gostaria de agradecer a todos que me ajudaram de forma direta ou indireta nesta trajetória.

Conteúdo

1	Introdução	4
1.1	Objetivos	4
1.2	Empresa Fácil Soluções Tecnológicas	4
2	Fundamentação Teórica	5
2.1	<i>Internet</i> das Coisas	5
2.2	Microcontroladores	5
2.3	Comunicação Serial	6
2.4	Comunicação RS-232	6
2.5	Linguagem PHP	6
3	Atividades do Estágio	8
3.1	Balança Inteligente	8
3.1.1	Projeto Eletrônico	8
3.1.2	Alimentação	9
3.1.3	Comunicação (MAX232N)	10
3.1.4	Microcontrolador	11
3.1.5	Gravação do programa	12
3.1.6	Circuito Impresso	13
3.1.7	Melhorias	13
3.1.8	Testando o circuito	14
3.2	Alarme	15
3.2.1	Apresentação	15
3.2.2	Definição do Projeto	15
3.2.3	Projeto Eletrônico	15
3.2.4	Alimentação	16
3.2.5	Microcontrolador	17
3.2.6	Amplificador de Áudio	17
3.2.7	Circuito impresso	18
3.2.8	Testando o circuito	18
3.3	Fechadura Eletrônica <i>FKnox</i>	20
3.3.1	Apresentação	20
3.3.2	Definição do Projeto	21
3.3.3	Esquemáticos e circuitos impressos	22
3.3.4	Problemas encontrados e suas soluções	24
4	Conclusões	30

Lista de Figuras

1	Esquema geral	8
2	Circuito do regulador de tensão	9
3	<i>Jack DC</i> A e a chave on/off B.	9
4	Balança	10
5	Ilustração do conector DB9 utilizado na balança	10
6	Esquemático para o funcionamento do MAX232	11
7	Pinout DB9	11
8	Esquemático do ESP	12
9	Conversor USB para TTL	12
10	<i>Layout</i> do circuito da balança	13
11	PCB montado pelo autor	13
12	Circuito montado com o novo <i>layout</i> projetado pelo autor também.	14
13	Pagina de configuração	14
14	Resultado mostrando a requisição e na url tem a informação do peso.	15
15	Esquema geral do circuito do alarme	16
16	Circuito regulador de tensão do alarme.	16
17	Esquemático do circuito do microcontrolador	17
18	Alto falante de 8 W	17
19	Esquemático do amplificador	18
20	<i>Layout</i> da placa confeccionada.	18
21	Momento de acionamento do alarme.	20
22	Execução do programa do alarme.	20
23	Fechadura Knox de acesso único	21
24	Fechadura Knox de acesso duplo	21
25	Esquemático no módulo Knox.	23
26	PCB da fechadura Knox. Dimensões: 45mm x 70mm	24
27	(B): Esquemático do contato a seco e em (A): a pcb	24
28	Fontes de alimentação da Fechadura <i>Knox</i>	25
29	A: Botão para destravar a porta pelo lado interno. (B): leitor RFID acesso pelo lado externo. (C): Eletroímã	26
30	(A): acesso interno. Neste <i>case</i> 3D contém os dois módulos e a placa do contato a seco. (B): acesso externo. Este <i>case</i> 3D externo engloba o leitor de cartão RFID e os <i>leds</i> , elementos que são ligados no modulo que está do outro lado por meio de um cabo	27

31	Utilização do espectro	28
32	Alterações do Knoxx para tentar solucionar o problema de conexão do Knoxx	29
33	(A): Resultado do knoxx com o <i>layout</i> antigo; neste caso o sinal de WiFi da rede a ser conectada não é encontrada.(B): Resultado do knoxx com o novo <i>layout</i> ; nessa caso o micro-controlador detectou a rede, conseguiu se comunicar e assim funcioanar normalmente.	29

1 Introdução

Neste relatório são apresentadas as atividades executadas pelo autor durante o Estágio Supervisionado realizado na empresa Fácil Soluções Tecnológicas no setor de Pesquisa e Desenvolvimento. Dentre as diferentes atividades executadas ao longo do estágio destacam-se a construção de circuitos impressos de projetos já existentes, bem como a melhoria e solução de demandas atuais e também o desenvolvimento de novos projetos para a empresa. As demandas solicitadas são relacionadas à integração de dispositivos aos sistemas da empresa.

1.1 Objetivos

O Estágio Supervisionado teve como objetivo produzir, desenvolver e melhorar dispositivos eletrônicos para serem utilizados tanto internamente quanto para atender a demandas externas, relativas aos produtos da empresa. Para executar as atividades foi necessário desenvolver as seguintes competências:

- Execução de processos para produção de circuitos impressos;
- Estudo e compreensão de comunicação *wireless* utilizando microcontroladores;
- Desenvolvimento de *firmware* para microcontroladores;
- Estudo e compreensão de circuito para acionamentos de dispositivos externos, como também de circuitos para aquisição de dados;
- Integração de circuitos eletrônicos com *softwares* em linguagem PHP;
- Identificação de problemas nos produtos e desenvolvimento de soluções viáveis.

1.2 Empresa Fácil Soluções Tecnológicas

A empresa Fácil Soluções Tecnológicas surgiu no mercado no ano de 2005, atuando no setor de tecnologia da informação, oferecendo soluções tecnológicas para o gerenciamento e controle de consignados.

Para conseguir o certificado da ISO9001, que é destinado à segurança da informação, foi necessário um sistema de controle de acesso aos recintos da empresa. Foram buscadas soluções no mercado que solucionassem este acesso,

porém não foi encontrado um sistema que atendesse às necessidades da empresa. Outro sistema requerido foi um sistema de alarme para sinalizar aos funcionários o horário do ponto; esses dois sistemas corresponderam às demandas internas. Com relação às demandas externas, a empresa, com o intuito de ampliar sua gama de atuações, expandiu sua gama de produtos e desenvolveu um sistema para atuar em bares e restaurantes chamado de *Facileat* (permitindo o gerenciamento das vendas em tais tipos de estabelecimentos).

Visando automatizar o sistema voltado para restaurantes e solucionar a demanda de controle de acesso da empresa foi passado essas tarefas para o setor de pesquisa e desenvolvimento da empresa, tais demandas foram repassadas ao setor de pesquisa para desenvolver tanto a parte de *software* como também a parte de *hardware*. A integração desses dispositivos aos sistemas da empresa são feitas por meio da *Internet*. Essa integração é dita como *Internet das coisas (Internet of Things) (IoT)*. Essas demandas foram solicitadas e boa parte delas já tinham um projeto realizado porém necessitavam de certo ajustes, bem como a montagem desses sistemas.

2 Fundamentação Teórica

Nesta seção serão apresentados os conceitos e ferramentas que foram utilizados durante atividades descritas neste Relatório.

2.1 *Internet das Coisas*

O termo IoT é uma analogia ao sistema *Web* nos dias de hoje que interliga as pessoas, e em que brevemente as "coisas" terão a habilidade de se comunicarem uma com as outras, proverão e usarão serviços, proverão dados e poderão reagir a eventos [1]. Outra analogia, agora mais técnica, é que IoT é vista como uma pilha de protocolos utilizados nos objetos inteligentes. Dessa forma, o conceito de IoT pode ser interpretado como dispositivos eletrônicos e sistemas de softwares que se comunicam através da *Internet*.

2.2 Microcontroladores

Um microcontrolador é um dispositivo que contém em um único circuito integrado alguns componentes, sendo os principais um processador, memória e periféricos de entrada e saída. Seu uso é dado em diversas aplicações

de sistemas embarcados, sendo comumente utilizados na automatização de processos e sistemas de controle de diversos dispositivos [2].

Devido ao seu baixo custo e à facilidade de reprogramação, são amplamente utilizados no mercado e também no desenvolvimento de protótipos visando algum produto final comercial.

Os microcontroladores são compostos por uma unidade de processamento, que é responsável por executar as instruções do programa de rotina, uma memória para armazenar dados como também as instruções a serem executadas e periféricos, portas de entrada e saídas de dados, tais como conversores analógico/digital e digital/analógico.

2.3 Comunicação Serial

Trata-se do processo que envia dados *bit por bit*. Existem vários protocolos para a comunicação serial, tais como, *RS-232*, *RS-422*, *RS-485*, *Universal Serial Bus*, *Firewire*, etc [4]. Essa comunicação pode ser realizada por meio de 3 fios que são: *RX* que é o receptor de dados; *TX* que é o transmissor; e a referência de de tensão *GND*.

2.4 Comunicação RS-232

Uma das formas mais empregadas para enviar dados de uma máquina a um computador, e vice-versa, corresponde a uma linha de comunicação serial sendo a mais conhecida a RS-232. Foi criado para ser usado com as antigas máquinas de teletipo, mas devido à sua simplicidade e versatilidade tornou-se um padrão que até hoje é adotado em aplicações de curta distância, principalmente as que envolvem a comunicação entre as máquinas (sensores e atuadores) e os computadores que as controlam [3].

O protocolo RS-232 transmite as informações por meio de 10 bit, em que primeiro bit é de *START*, a informação é transmitida pelo bits de 2-8, o bit de número 9 é o bit de paridade que serve para verificar se a informação foi enviada de forma correta e por fim o bit 10 é o bit de *STOP*.

2.5 Linguagem PHP

O PHP (um acrônimo recursivo para PHP: *Hypertext Preprocessor*) é uma linguagem de *script open source* de uso geral, muito utilizada, e es-

pecialmente adequada para o desenvolvimento *Web*, que pode ser embutida dentro do linguagem de marcação de hipertexto (*Hypertext Markup Language*) (HTML) [5].

3 Atividades do Estágio

Nesse capítulo será descrito como foram executados os projetos e atividades durante o estágio supervisionado, entre Março e Setembro de 2018. Os três projetos realizados foram: Balança inteligente; Fechadura Eletrônica Knox; e o Alarme para indicar os horários de bater o ponto aos funcionários.

3.1 Balança Inteligente

A Balança Inteligente é utilizada para enviar ao sistema do *FacilEat*, de forma automática, o peso de alimentos consumidos que são cobrados ao cliente por meio dessa variável.

Assim, ao solicitar ao sistema o valor desse peso, o sistema envia esta requisição ao dispositivo que está acoplado à balança, que por sua vez responde com esta informação. Na Figura 1 tem-se o esquemático proposto para a balança

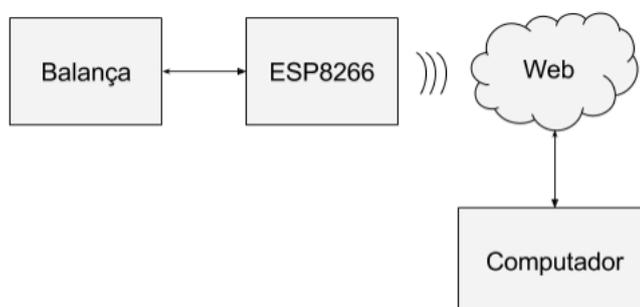


Figura 1: Esquema geral

3.1.1 Projeto Eletrônico

O circuito é dividido em blocos cujas funções são distintas. O circuito de alimentação responsável por fornecer o nível tensão correto para os componentes presentes no circuito; O microntrolador, responsável por se comunicar com o servidor *Web* e com a balança; e a parte do MAX-232 cuja função é converter o protocolo de comunicação da balança (*RS-232*) para o protocolo de comunicação do microntrolador (*TTL*). Esse projeto havia sido elaborado previamente na empresa, sendo repassada a produção do circuito e a tarefa de projetar um novo *layout* utilizando o CI MAX – 232 com encapsulamento tecnologia de montagem em superfície (*Surface Mounting Device*

ou Componente) (smd).

3.1.2 Alimentação

Apenas dois elementos precisam ser alimentados com tensões específicas que são: o *MAX-232* (*Texas Instruments*) e o microcontrolador ESP8266 12, ambos sendo alimentados com 3.3 V. O datasheet do *MAX-232* recomenda uma alimentação de 5 V e o mínimo de 4.5 V, mas para esse caso 3.3 V já é suficiente [7]. Sendo assim, o regulador de tensão do circuito é o AMS1117 de 3.3V, esse é o regulador presente na lista de componentes da empresa e por essa razão foi utilizado. O esquemático utilizado é apresentado na Figura 2

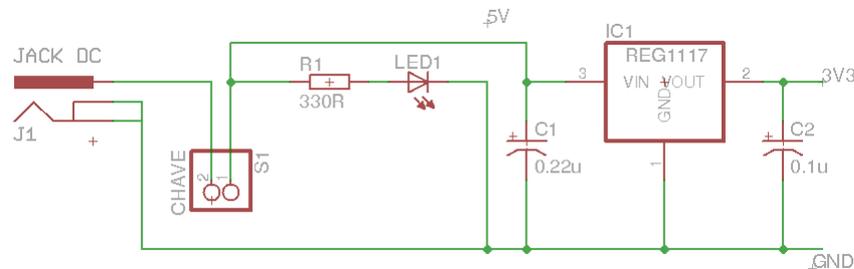


Figura 2: Circuito do regulador de tensão

O circuito tem um *Jack DC* como o da Figura 3 A. Esse conector serve para plugar uma fonte com uma alimentação podendo ser entre 5 V e 9 V. Há um *Led* e um resistor (330Ω) apenas para sinalizar que está ligado ou desligado o circuito e uma chave para ligar e desligar o circuito, 3 B.



Figura 3: *Jack DC* A e a chave on/off B.

Os capacitores estão nos valores recomendados pelo *datasheet* do regulador de tensão, que servem como filtros para diminuir oscilações de tensão que venham comprometer o funcionamento do circuito:

- Capacitor de entrada: 220 nF ;
- Capacitor de saída: 100 nF ;

3.1.3 Comunicação (MAX232N)

A balança utilizada no teste foi uma *ELGIN SA – 110*, ilustrada na Figura 4. A mesma possui uma saída serial com conector DB9 como o da Figura 5.



Figura 4: Balança



Figura 5: Ilustração do conector DB9 utilizado na balança

Para a comunicação com a balança, torna-se necessário um circuito conversor de níveis de tensão. O circuito que faz isso é o circuito do MAX232, que é apresentado na Figura 6.

Os capacitores podem ser utilizados nos valores de $0.1\mu F$ ou de $1\mu F$; Caso sejam utilizados capacitores eletrolíticos (com polaridade), a polaridade precisa seguir as polaridades dos capacitores dos circuitos de exemplo que são mostradas no esquemático da Figura 6 que é idêntico ao do *datasheet*. As ligações do MAX232 com o conector DB9 são

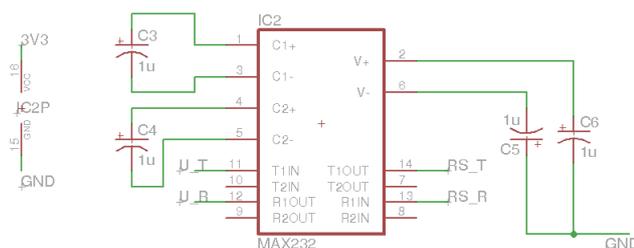


Figura 6: Esquemático para o funcionamento do MAX232

- PIN -13 MAX(R1IN) ————— PIN2 DB9(RX)
- PIN -14 MAX(T1OUT) ————— PIN3 DB9(TX)
- GND ————— PIN5 GND

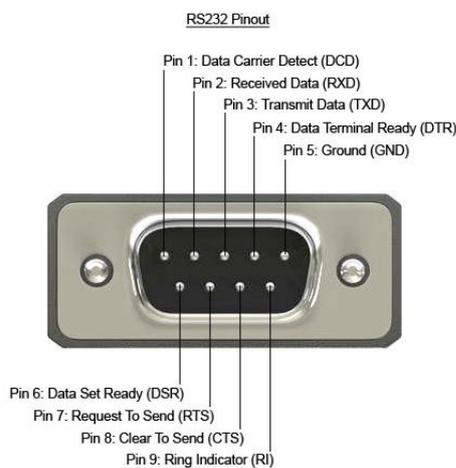


Figura 7: Pinout DB9

3.1.4 Microcontrolador

O circuito necessita da alimentação e de alguns componentes para dar melhor estabilidade ao seu funcionamento. Um capacitor é utilizado o mais próximo possível do microcontrolador. Para os pinos Reset, Chip Enable e GPIO (porta de entrada ou saída) 0 são utilizados resistores de *Pull-Up*. O pino GPIO15 possui um resistor de *Pull-Down*, pois o ESP8266 necessita de determinada configuração para o modo de inicialização que se deseja. A tabela a seguir representa esses modos:

Dessa forma, o seguinte esquemático para o microcontrolador é ilustrado na Figura 8

Assim, a chave entre o GPIO0 e o GND é utilizada para a gravação do sistema embarcado. Ao colocar GND no terra e ligar o microcontrolador ele

GPIO15	GPIO0	GPIO2	Modo	Descrição
L	L	H	UART	Modo de gravação
L	H	H	FLASH	Executa o programa
H	X	X	SDIO	Executa o programa no sd

Tabela 1: Modos do esp. H = Nível lógico alto, L= Nível lógico baixo, X = não importa

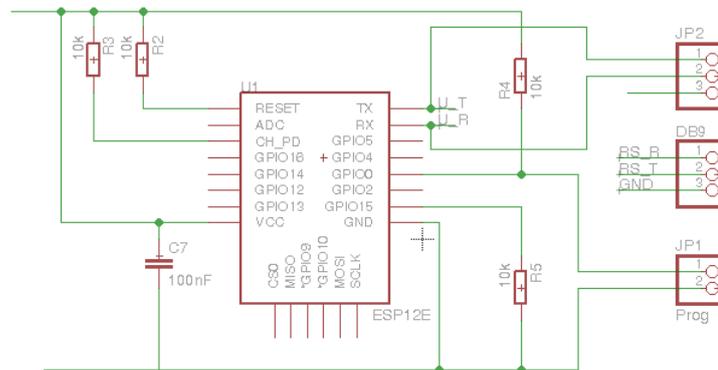


Figura 8: Esquemático do ESP

estará em modo de gravação.

3.1.5 Gravação do programa

Para gravar o programa da balança é necessário apenas de um conversor *USB para RS – 232TTL*, ilustrado na Figura 9



Figura 9: Conversor USB para TTL

A ligação deve ser a seguinte:

- Conversor RX — Tx *ESP*
- Conversor TX — RX *ESP*
- GND — GND *ESP*

O programa da balança está em linguagem C. A programação do sistema embarcado pode ser feita através da IDE do Arduino.

3.1.6 Circuito Impresso

O projeto da placa de circuito impresso é mostrado na Figura 10. A alimentação é feita pelo Jack DC. O cabo DB9 é conectado aos pinos RX, TX e GND que estão expostos.

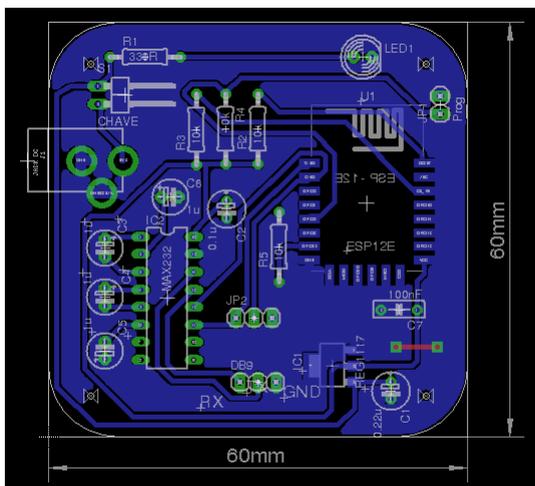


Figura 10: *Layout* do circuito da balança

Essa placa foi montada e o resultado é apresentado na Figura 11

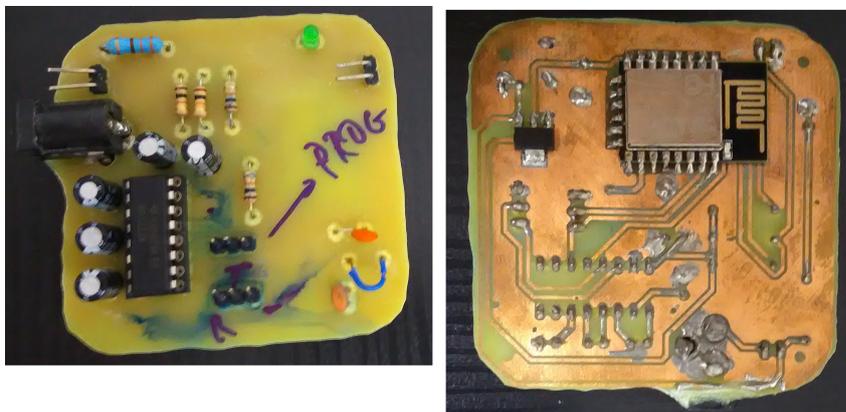


Figura 11: PCB montado pelo autor

3.1.7 Melhorias

Foi solicitada a mudança do *layout* para utilizar o CI *MAX – 232* com o encapsulamento smd. A nova placa, desenvolvida durante o estágio, tem menor dimensão. O resultado é apresentado na Figura 12.



Figura 12: Circuito montado com o novo *layout* projetado pelo autor também.

3.1.8 Testando o circuito

Ao ligar o dispositivo, um *Access Point* é criado para ajustar as configurações. A rede em questão pode ser acessada digitando-se o IP 10.10.10.1 no navegador, após isso 13.

reco MAC: 62:1:94:E:81:C

Entradas

1) ativo (-85)
2) inativo (-59)

Easy

000bcdec43

Senha: 0 0 0 0

IP: 0 0 0 0

Porta: 0 0 0 0

Host: 10.0.100.131

URL: /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/

Porta: 9600

Requisição: 5

Salvar



Figura 13: Pagina de configuração

Os campos para a configuração são os seguintes:

- Rede: rede WiFi na qual o microcontrolador vai se conectar;
- Senha: senha da rede WiFi na qual o microcontrolador vai se conectar;
- *Host*: servidor;
- Requisição: página que o microcontrolador vai acessar, além de ser o local em que o peso será salvo;

- *Baud rate*: *baude rate* da balança;
- *Enquiry*: caractere para requisição do peso; nesse caso é o valor 5.

O resultado esperado é apresentado na Figura 14. Pelo terminal foi verificado as requisições feitas ao servidor e no final dessa requisição é enviado o valor do peso que o circuito adquiriu.

```
HTTP/1.1" 404 558 "-" "-"
10.0.100.200 - - [24/Jul/2018:14:25:24 -0300] "GET /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/9b5c544994750606a26ee5adff56898f?peso=<IIIII>
HTTP/1.1" 404 558 "-" "-"
10.0.100.200 - - [24/Jul/2018:14:25:25 -0300] "GET /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/9b5c544994750606a26ee5adff56898f?peso=<IIIII>
HTTP/1.1" 404 558 "-" "-"
10.0.100.200 - - [24/Jul/2018:14:25:26 -0300] "GET /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/9b5c544994750606a26ee5adff56898f?peso=<00028>
HTTP/1.1" 404 558 "-" "-"
10.0.100.200 - - [24/Jul/2018:14:25:27 -0300] "GET /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/9b5c544994750606a26ee5adff56898f?peso=<00028>
HTTP/1.1" 404 558 "-" "-"
10.0.100.200 - - [24/Jul/2018:14:25:28 -0300] "GET /restaurantes/facileat_demo/configuracoes/balanca/get_dados/balanca_id/9b5c544994750606a26ee5adff56898f?peso=<00028>
```

Figura 14: Resultado mostrando a requisição e na url tem a informação do peso.

Onde $\langle IIIII \rangle$ significa que o peso está instável e $\langle 00028 \rangle$ é o peso em gramas, neste caso o valor correspondente são $28g$.

3.2 Alarme

3.2.1 Apresentação

As informações a respeito do circuito eletrônico, programação e montagem do circuito do alarme estão presentes nesta seção.

3.2.2 Definição do Projeto

O alarme apresentado nesta seção é utilizado na empresa com o intuito de avisar, por meio sonoro, os horários de chegada, saída e o horário de almoço para os funcionários da empresa. Evitando dessa forma que os funcionários percam os horários certos de bater o ponto.

O sistema do alarme funciona da seguinte forma, o microcontrolador que se conecta à rede e faz uma requisição ao servidor *ntp* para obter a informação da hora, evitando assim que a contagem de tempo se atrase ou se adiante. Com a informação da hora o alarme será acionado apenas em determinadas horas.

3.2.3 Projeto Eletrônico

O circuito é dividido em três partes (Este projeto foi totalmente elaborado pelo autor):



Figura 15: Esquema geral do circuito do alarme

- Alimentação;
- Microcontrolador;
- Amplificador de áudio;

3.2.4 Alimentação

O circuito deve alimentar dois componentes, que são o microcontrolador *ESP8266 – 12E* e o CI responsável por amplificar o sinal de áudio. Dessa forma é suficiente que a alimentação seja de 5 V. Para alimentar o *ESP* é utilizado o regulador de tensão AMS1117 de 3.3 V.

Neste contexto, o esquemático da Figura 16 foi utilizado para o regulador de tensão. O esquemático proposto dispõe de uma chave para ligar e desligar a alimentação e um *led* para indicar se o circuito está ligado ou não. Tem-se dois capacitores como filtros para alimentação. Uma fonte de 5 V vai ser responsável por alimentar todo circuito. Os pinos de acesso são bornes.

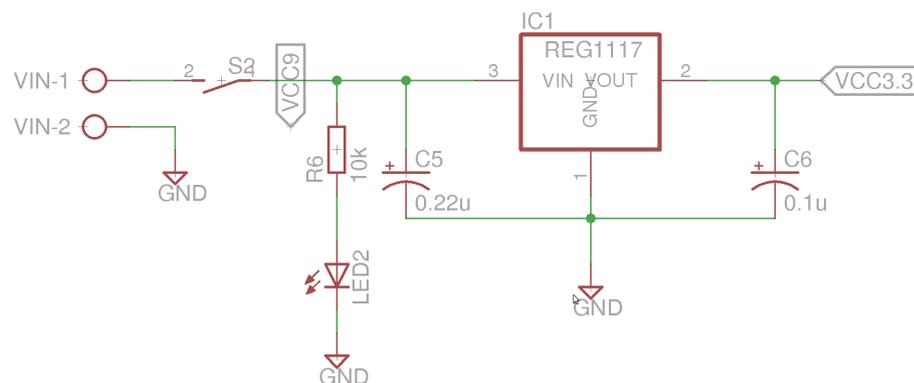


Figura 16: Circuito regulador de tensão do alarme.

3.2.5 Microcontrolador

A configuração do microcontrolador manteve-se a mesma. Entretanto foram adicionados duas GPIOs uma para saída do sinal de áudio (GPIO12) e outra para um led (GPIO14), que será utilizado para indicar em que ponto do programa está sendo executado. Também foi adicionado um botão de *reset*. O esquemático proposto é apresentado na Figura 17

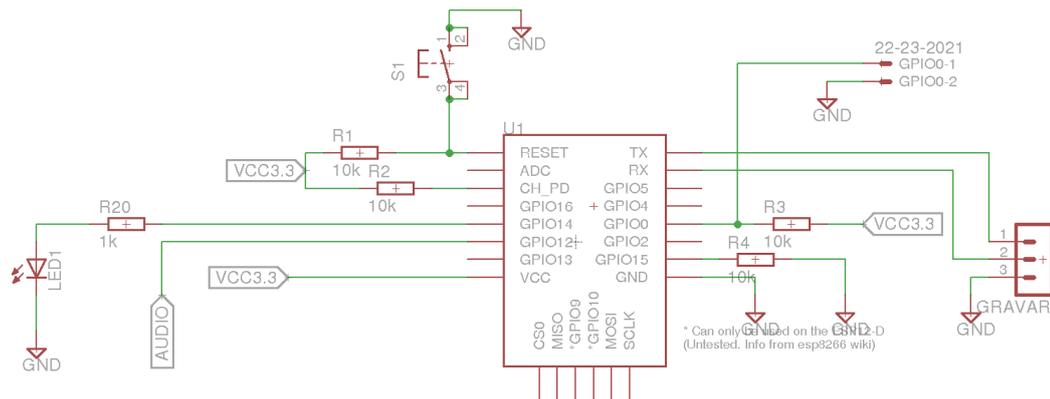


Figura 17: Esquemático do circuito do microcontrolador

3.2.6 Amplificador de Áudio

A saída do microcontrolador não fornece a potência suficiente para acionar um alto falante de 8W, que é utilizado na empresa (e é ilustrado na Figura 18). Desta maneira, foi necessário utilizar o CI LM386N¹, que se trata de um amplificador de áudio, de modo a permitir a correta operação do alto falante considerado.



Figura 18: Alto falante de 8 W

O circuito implementado a partir do LM386 é apresentado na Figura 19. O potenciômetro logarítmico de entrada serve para regular o volume.

¹<http://www.ti.com/lit/ds/symlink/lm386.pdf>

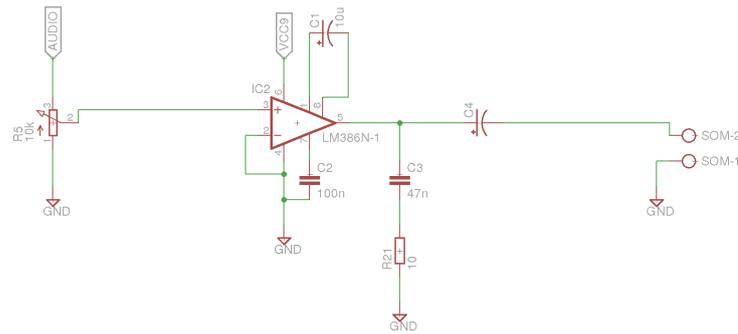
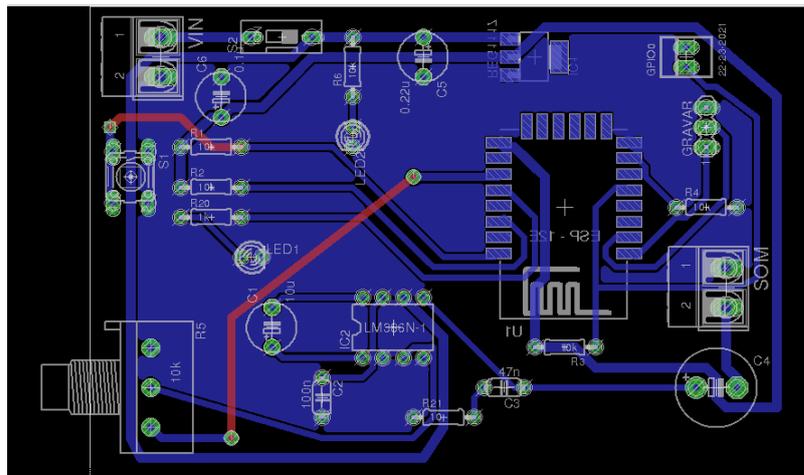


Figura 19: Esquemático do amplificador

3.2.7 Circuito impresso

Na placa de circuito impressa confeccionada, há bornes para ligar a alimentação e o alto falante. Os pinos do GND, RX e TX estão expostos para programar o microcontrolador. O pino do GPIO0 também está exposto para alternar entre modo de programação e modo para executar do programa.

Figura 20: *Layout* da placa confeccionada.

3.2.8 Testando o circuito

Após soldar os componentes e gravar o programa, a próxima etapa envolveu o teste do programa. O programa executa a seguinte rotina:

- Conecta-se a rede de WiFi estabelecida no programa. O *led* indicador vai piscar de forma lenta.
- Caso encontre a rede, o microcontrolador vai tentar se conectar. O *led* indicador vai piscar um pouco mais rápido.

- Uma vez conectado, o led indicador ficará aceso e o microcontrolador vai requisitar a hora do servidor *ntp*.
- A hora é gravada em uma variável e comparada às horas preestabelecidas para que o alarme seja acionado.
- Caso o horário atual não seja igual às horas preestabelecidas, o programa volta para o ponto de requisição da hora ao servidor *ntp*. Dessa forma, evitam-se atrasos e adiantamentos do horário, pois a todo instante é requisitado o horário do servidor.
- Caso o horário atual seja iguais às horas preestabelecidas a função de execução da música é chamada. Esta função envia um sinal pela GPIO12 em que cada nota é reproduzida por através de uma modulação por largura de pulsos (*Pulse Width Modulation*) (PWM). No *ESP8266 – 12E* não há saídas analógicas. Dessa forma as frequências das notas são reproduzidas por *PWM*. A função do *PWMf* para reproduzir as notas possui três parâmetros:

1	<code>PWMf(Pin, Nota, Tempo);</code>
---	--------------------------------------

onde *Pin* é a GPIO escolhida, *Nota* é o valor da frequência e *Tempo* é a duração da nota, ou seja, é o tempo em que a nota vai ser executada. Assim, listando-se todas as notas e seus respectivos tempos de duração é possível reproduzir a música desejada ao executá-las em sequência. As horas preestabelecidas são os horários de chegada e saída e os horários de almoço, quais sejam:

- 08:00 h (chegada);
 - 12:00 h (almoço);
 - 14:00 h (retorno do almoço);
 - 18:00 h (saída);
- Ao executar a música, o microcontrolador reinicia, voltando ao início do programa. Isso foi feito para evitar barulhos após a execução da função de tocar a música. Assim a GPIO 12 do alto falante é desativada e nenhum sinal é amplificado e passado para o alto falante. Mesmo zerando a saída da GPIO ainda restava algum pequeno sinal que era amplificado e o alto falante permanecia emitindo ruídos. A execução do programa é mostrada nas Figuras 22 e 21.

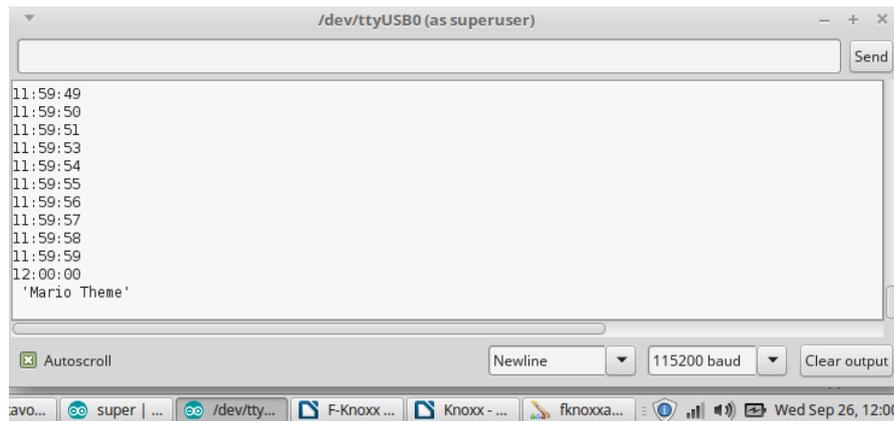


Figura 21: Momento de acionamento do alarme.

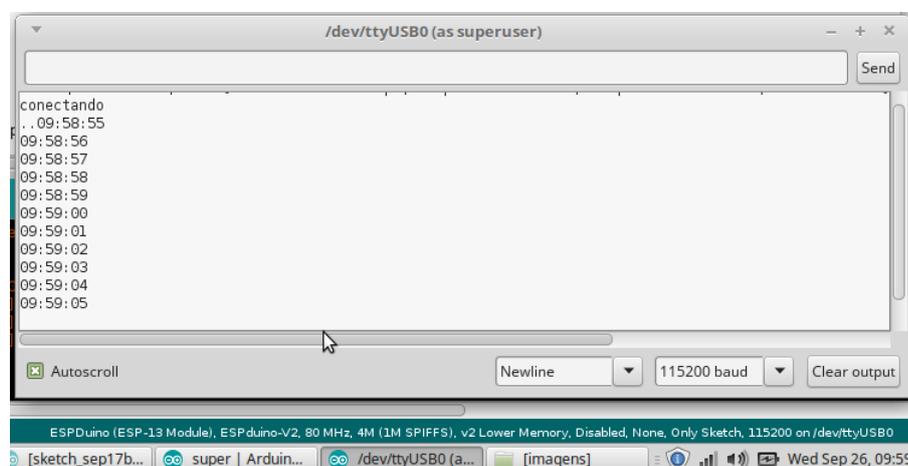


Figura 22: Execução do programa do alarme.

3.3 Fechadura Eletrônica *FKnox*

As fechaduras eletrônicas que serão detalhadas nesta seção já estavam instaladas e em operação na empresa. Porém, vários problemas surgiram e foi necessário realizar alterações de hardware e de software para correção dos problemas.

3.3.1 Apresentação

O sistema *Fknox* é responsável por controlar o acesso dos funcionários aos recintos da empresa. Registra-se, assim, qual pessoa teve acesso e em que dia e hora isso aconteceu. Este projeto já havia sido implementado na empresa, contudo durante o período deste estágio o sistema apresentou vários problemas que exigiram mudanças tanto no hardware quanto no software.

3.3.2 Definição do Projeto

O sistema tem duas opções. A primeira opção é um sistema de acesso único com um botão. A segunda opção é um sistema de acesso duplo sem botão. A Figura 23 representa o sistema de acesso único e a Figura 24 representa o sistema de acesso duplo.

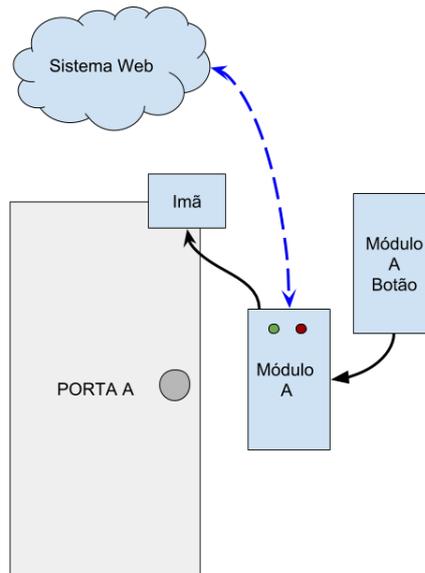


Figura 23: Fechadura Knoxx de acesso único

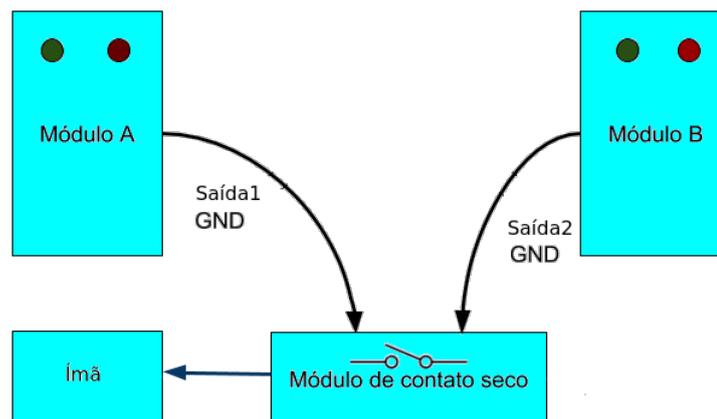


Figura 24: Fechadura Knoxx de acesso duplo

No projeto de acesso único tem-se os seguintes elementos:

- Servidor
- Eletroímã

- Botão
- Módulo *Knox*

No projeto de acesso duplo tem-se os seguintes elementos:

- Servidor Web
- Eletroímã
- Contato a seco
- 2 Módulos *Knox*

Os elementos citados possuem as seguintes funções:

- Servidor Web: Responsável por verificar a lista de cartões e se esse cartão tem acesso ou não, bem como registrar a data e hora de quem fez a requisição.
- Eletroímã: Responsável por travar a porta quando acionado e por destravar quando desligado.
- Módulo: Responsável por se comunicar com o servidor web, sinalizar através dos *leds* como também mandar o sinal para desligar e ligar o eletroímã.
- Contator a seco: é uma porta lógica feita com diodos para que os dois módulos possam liberar o eletroímã. Possui um relé para acionar o eletroímã ou desliga-lo. Responsável também por alimentar os dois módulos. Assim, todo o sistema necessita apenas de uma alimentação.
- Botão: O botão é responsável por liberar o acesso pelo lado interno da porta.

3.3.3 Esquemáticos e circuitos impressos

O esquemático do módulo apresentado na Figura 25;

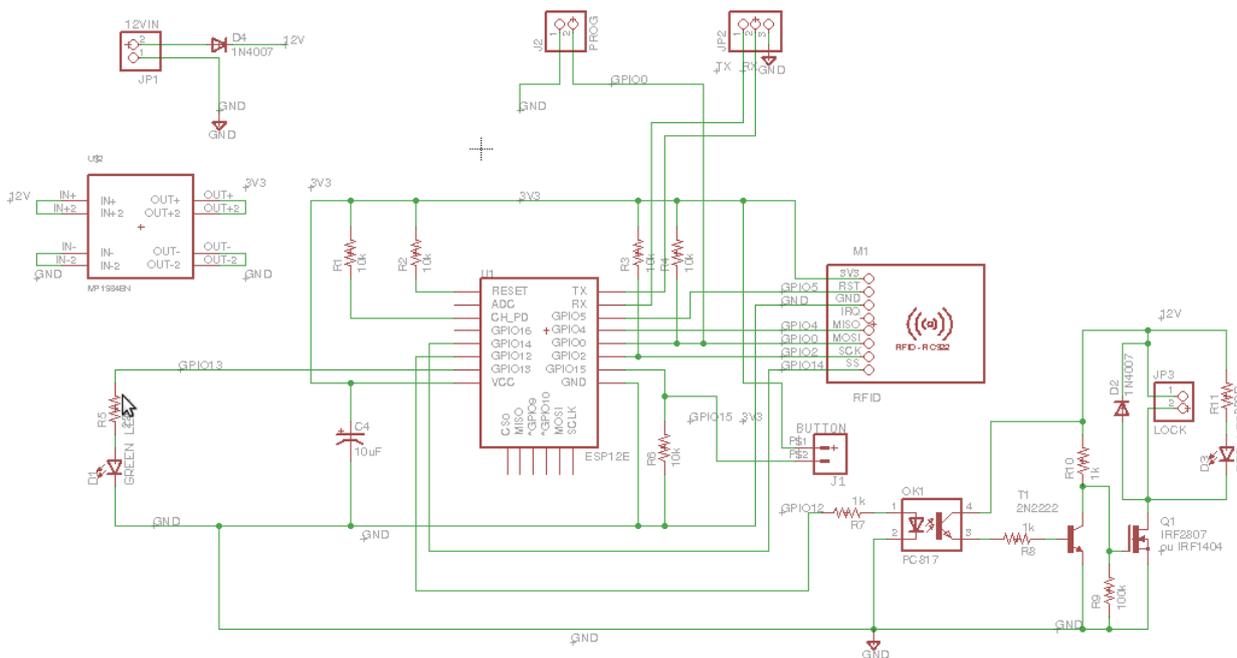


Figura 25: Esquemático no módulo Knnox.

Nesse circuito é utilizado como regulador de tensão um conversor *Buck*. Como a tensão de alimentação é 12 V, os reguladores de tensão lineares aquecem muito e sua vida útil é reduzida. Portanto, foi escolhido o elemento *MP1584EN* como regulador.

O diodo *D2* em paralelo com a saída LOCK (saída para o ímã) serve para proteger o restante do circuito no momento em que o ímã for descarregado.

O acionamento do ímã é composto por um fotoacoplador, um transistor bipolar e um mosfet. O fotoacoplador serve para isolar a GPIO do circuito de acionamento. O transistor entra em condução quando o GPIO está em nível lógico alto, fazendo com que o nível lógico de tensão na base do mosfet seja baixo e assim o mosfet não entra em condução - desativando, assim, o eletroímã. Caso o nível lógico da GPIO esteja baixo, o transistor não entra em condução e o nível de tensão na base do mosfet vai ser o resultado do divisor resistivo entre *R10* e *R9*, ativando o mosfet e consequentemente ativando o ímã.

O circuito do botão é um resistor de *Pull Down* que está ligado na GPIO no microcontrolador. O botão estão ligado na GPIO e em 3.3 V; assim, quando acionado ele vai fazer com que o nível de tensão na GPIO seja de 3.3 V quando não está acionado o nível de tensão é 0 V. Na Figura 26 é ilustrado o circuito da fechadura *Knnox*.

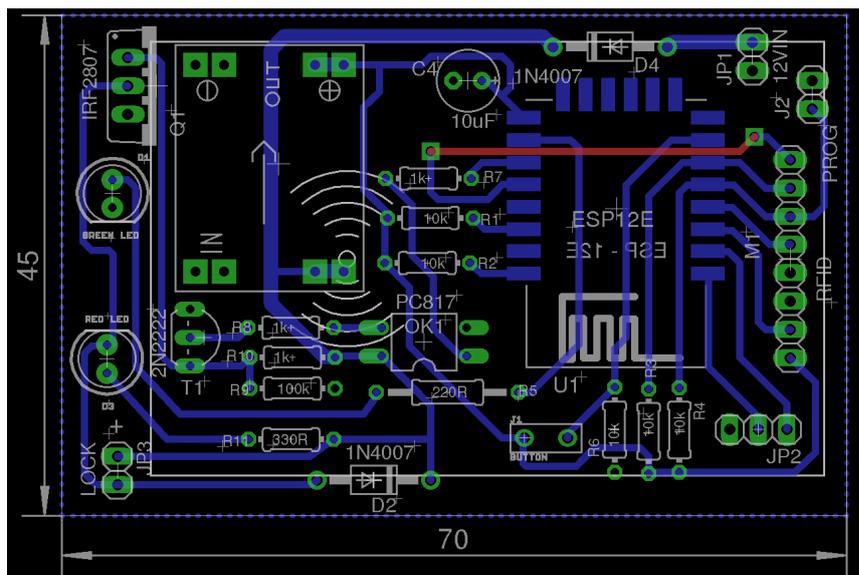


Figura 26: PCB da fechadura Knoxx. Dimensões: 45mm x 70mm

O esquemático do contato a seco e a sua pcb são ilustrados na Figura 27 A e B. Os conectores $X1 - 1/ - 2$ são a entrada da fonte de alimentação, os conectores $X2 - 1/ - 2$ e $X5 - 1/ - 2$ são saídas para alimentar os módulos e os conectores $X3 - 1/ - 2$ e $X4 - 1/ - 2$ são as entradas do sinal *LOCK* de cada um dos módulos, ou seja, são as entradas da porta lógica *OR* devidos aos diodos $D1$ e $D2$. O diodo $D3$ é um diodo de roda livre para o relé. Os conectores $X6 - 1/ - 2$ servem para coenctar o íma. Nesse projeto falta um diodo de roda livre para o ímã. Esse problema será detalhado posteriormente.

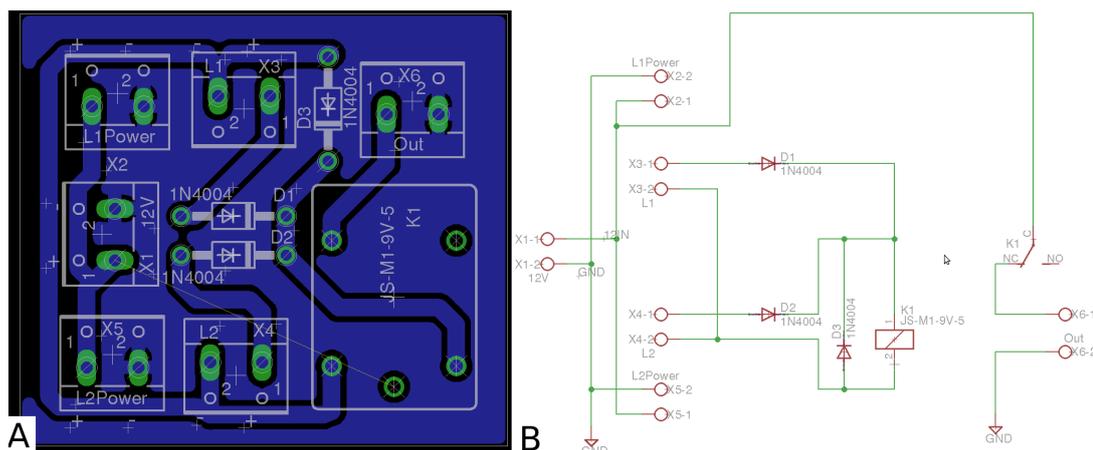


Figura 27: (B): Esquemático do contato a seco e em (A): a pcb

3.3.4 Problemas encontrados e suas soluções

Todas as fechaduras são alimentadas pelas fontes ilustradas na Figura 28. A fonte do canto esquerdo superior da Figura 28 foi o primeiro tipo de fonte

instalada, possui saída de 13.8 V com capacidade máxima de fornecer 3A e entrada para bateria. As fontes funcionaram por um longo tempo, e após uma oscilação de energia elétrica da rede duas delas queimaram e foram substituídas por fontes *Tektop* de 12 V com a capacidade máxima de 4 A.



Figura 28: Fontes de alimentação da Fechadura *Knox*

Após a instalação de novos roteadores de WiFi surgiu um problema no funcionamento das fechaduras eletrônicas *Knox* de acesso único (no total, há três fechaduras de acesso único instaladas nas dependências da empresa). As fechaduras estavam ligando e desligando o ímã constantemente. A primeira suspeita foi com relação à oscilação da alimentação. Foram realizados testes de tensão e corrente nas fontes e o comportamento foi normal. Eliminou-se, assim, a possibilidade do problema estar ligado à alimentação das mesmas. A segunda suspeita era a conexão com o servidor e, apesar de a fechadura estar apresentando falhas durante todo o tempo, a porta estava travando e destravando ao se passar o cartão - portanto, o Knox se comunicava. A terceira possibilidade foi algum defeito de hardware, algum componente com problema ou mau contato na placa devido ao tempo. Então foi feita uma nova placa e o problema persistiu. Após algum tempo foi identificado que a porta estava com esse problema pois o microcontrolador estava detectando o acionamento do botão o tempo todo. O botão foi trocado por um novo e o problema ainda persistiu. Após algumas observações, foi percebido que a origem do problema era interferência eletromagnética: o fio do botão que precisa atravessar a parede funcionava como uma antena e a todo pulso do sinal o fio captava o microcontrolador identificava isso como uma borda de subida consequentemente acionando o botão.



Figura 29: A: Botão para destravar a porta pelo lado interno. (B): leitor RFID acesso pelo lado externo. (C): Eletroímã

Para solucionar esse problema foram feitas duas modificações: uma via software e a outra via hardware. O código do botão funcionava da seguinte maneira: o botão opera por interrupção; a cada instante ele verifica se o botão foi acionado ou não. Essa detecção é feita por borda de subida (transição do nível lógico baixo para o alto). Após essa detecção ele zera o tempo em que o botão foi acionado e chama outra função. Nessa outra função é utilizado de novo o tempo em que o botão foi acionado e caso o tempo após a interrupção passe de 2000 ms, a função para destrancar a porta é chamada. O problema, é que devido a interferência, o botão ficava acionando o tempo todo a interrupção, precisando assim de um filtro mais seletivo para identificar o acionamento do botão. Tendo isto em vista, a seguinte estratégia foi executada: adicionou-se um laço de comparação dentro da função que é ativada após a interrupção e esse laço de comparação verifica duas condições: uma é a condição do tempo de 2000 ms e a outra é o nível da GPIO15 (porta que está ligado o botão); caso o nível lógico seja alto também, a função de destrancamento da porta é chamada. As duas condições precisam ser satisfeitas para destrancar a porta.

A modificação de hardware foi utilizar um fio blindado para fazer a ligação entre a placa e o botão. Com ambas as modificações conseguiu-se eliminar a influência da interferência eletromagnética, e

a porta só destranca quando de fato o botão for acionado.

Outro problema aconteceu no *Knox* de acesso duplo que pode ser observado na Figura 30. Esse sistema montado estava travando após a primeira liberação do acesso, ou seja, após o ímã descarregar. O leitor de cartão não funcionava mais após esse evento. O problema aparentemente estava no leitor de cartão de RFID. Entretanto, após fazer o teste de desconectar o eletroímã, o sistema funcionava normalmente sem travar por inúmeras vezes. Assim, observando novamente esquemático da Figura 27, percebeu-se falta um diodo para descarregar o eletroímã. Assim, no momento em que o eletroímã é desativado não havia por onde descarregá-lo. Como o terra das placas de contato seco e dos dois módulos são interligados, um sinal era repassado pelo terra, causando interferência no leitor de cartão fazendo com que ele parasse seu funcionamento. Esse problema foi solucionado colocando um diodo em paralelo com os conectores $X6 - 1/ - 2$. Após essa modificação o sistema funcionou sem travar.

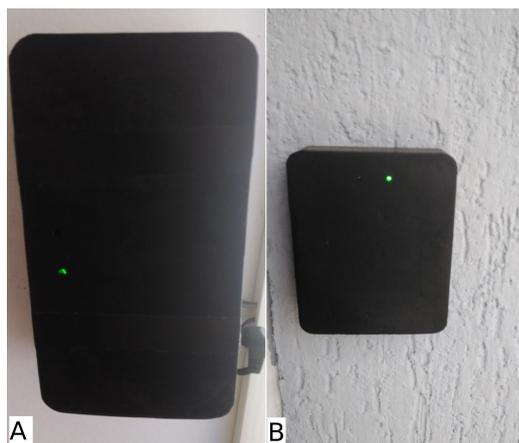


Figura 30: (A): acesso interno. Neste *case* 3D contém os dois módulos e a placa do contato a seco. (B): acesso externo. Este *case* 3D externo engloba o leitor de cartão RFID e os *leds*, elementos que são ligados no módulo que está do outro lado por meio de um cabo

Ao longo do tempo algumas placas precisaram ser substituídas e outras o microcontrolador precisou ser substituído. E após isso algumas tiveram problema para se conectar com a rede WiFi. Ainda não foram identificadas as causas desta falha, porém as seguintes possibilidades podem contribuir para o problema:

- Interferência do leitor de cartão RFID, interferindo no sinal de Wi-fi. O problema não persiste após a retirada do cartão.
- Objetos na frente ou em volta do módulo que pioram a qualidade do sinal. O módulo fica instalado dentro de uma caixa no interior da parede

ou dentro do *case 3D*.

- Diversos sinais de Wi-fi na área da empresa. Na Figura 31 pode-se observar a ocupação espectral na faixa de frequências investigada. E em alguns pontos da empresa foram detectados até 52 redes de WiFi.

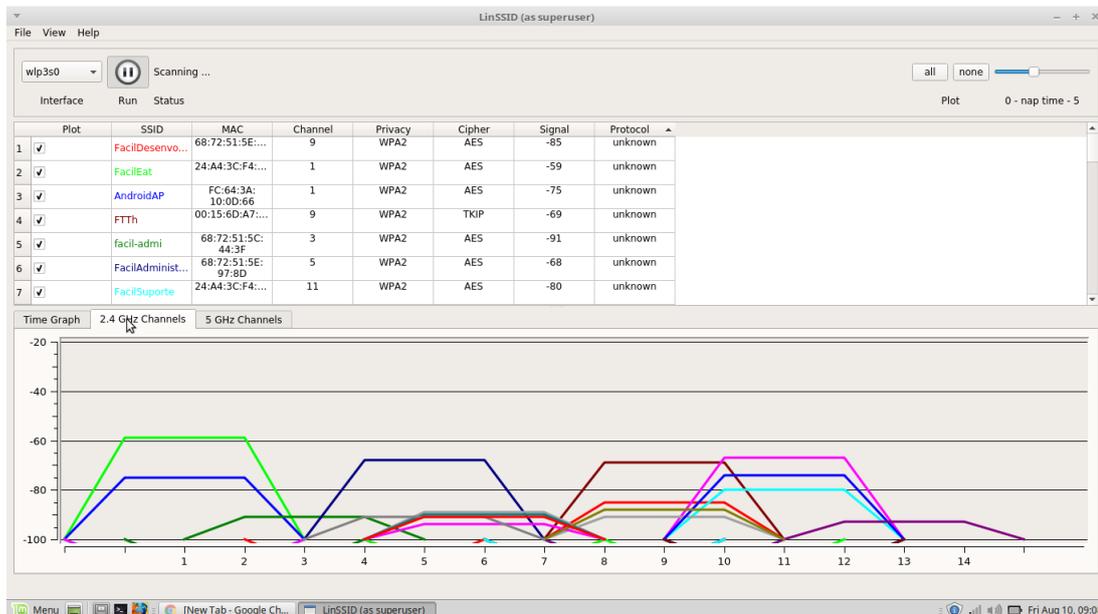


Figura 31: Utilização do espectro

A tentativa de solucionar esse problema foi desenhar um novo *layout* para a fechadura. As modificações foram duas:

- Deslocar o ESP para deixar a sua antena mais exposta. No modelo anterior o microcontrolador fica por baixo do leitor de RFID.
- Desenhar uma trilha na própria placa que vai servir de antena externa para o ESP.

Em contrapartida, para conseguir realizar essas modificações o tamanho da placa aumentou. Suas dimensões passaram a ser 53×84 mm. Na Figura 32 as intervenções realizadas na placa podem ser observadas.

O resultado é ilustrado na Figura 33. Como pode-se observar, o sinal melhorou consideravelmente. As dimensões utilizadas para desenhar a trilha foram retiradas do próprio formato da antena do microcontrolador *ESP8622-12F*, sendo medidas e reproduzidas no desenho da pcb.

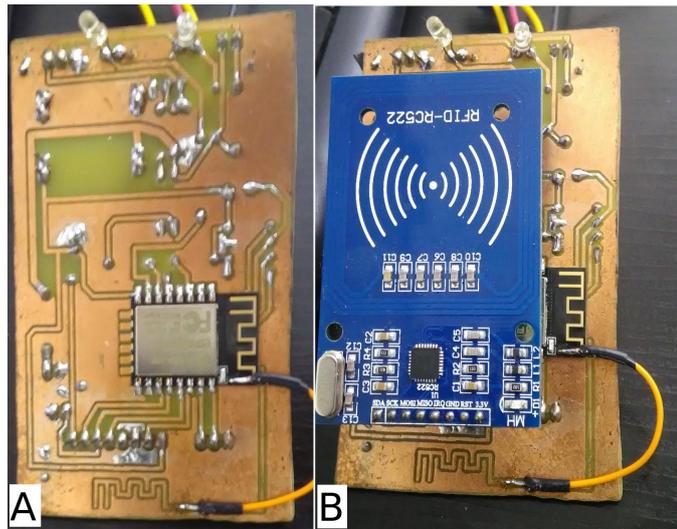


Figura 32: Alterações do Knoxx para tentar solucionar o problema de conexão do Knoxx

```

---- Forte Knoxx ----
scandone
scandone
Nome da rede: b'FacilDesenvolvimento'
Potencia do sinal -64

Nome da rede: b'ESP_0750F6'
Potencia do sinal -73

reconnect
Conectando:
ESSID: facil-admi
Password: ██████████
scandone

MicroPython v1.9.2-123-gbdc6e86-dirty on 2017-10-09; ESP module with ESP8266
Type "help()" for more information.
>>> Status: 3
scandone
no facil-admi found, reconnect after 1s
reconnect
Status: 3
scandone
no facil-admi found, reconnect after 1s
reconnect
Status: 3
scandone
no facil-admi found, reconnect after 1s

A

---- Forte Knoxx ----
scandone
scandone
Nome da rede: b'FacilDesenvolvimento'
Potencia do sinal -61

Nome da rede: b'FTTH'
Potencia do sinal -84

Nome da rede: b'Easy'
Potencia do sinal -74

Nome da rede: b'FacilSuporte'
Potencia do sinal -85

Nome da rede: b'facil-admi'
Potencia do sinal -75

Nome da rede: b'ESP_0750F6'
Potencia do sinal -67

Nome da rede: b'RosaCorrida'
Potencia do sinal -92

reconnect
Conectando:
ESSID: facil-admi
Password: ██████████
scandone

MicroPython v1.9.2-123-gbdc6e86-dirty on 2017-10-09; ESP module with ESP8266
Type "help()" for more information.
>>> Status: 3
scandone
state: 0 -> 2 (b0)
state: 2 -> 3 (0)
state: 3 -> 5 (10)
add 0
aid 10
cnt

connected with facil-admi, channel 11
dhcp client start...
Status: 3
ip:10.0.100.181,mask:255.255.255.0,gw:10.0.100.1
Status: 1
Status: 1
Retorno:
F0E06A20
Host:
10.0.15.15
Path:
knoxx/knoxx.php/autentica/F0E06A20/R91w0ZykbjUkLtk0bLW

Trig

Conteudo:
{"status":"true","openTime":"3000","blinkTime":300,"blinkRepetitions":3}
pm open,type:2 0
Status: 1

B

```

Figura 33: (A): Resultado do knoxx com o *layout* antigo; neste caso o sinal de WiFi da rede a ser conectada não é encontrada.(B): Resultado do knoxx com o novo *layout*; nessa caso o microcontrolador detectou a rede, conseguiu se comunicar e assim funciona normalmente.

4 Conclusões

Nesse relatório foram apresentadas as principais atividades realizadas pelo autor no estágio supervisionado na empresa Fácil Soluções Tecnológicas, no setor de Pesquisa e Desenvolvimento. As atividades realizadas exigiram conteúdos que normalmente não são abordados no meio acadêmico, como também proporcionou o desenvolvimento de um raciocínio para identificar os problemas e encontrar soluções viáveis. O desenvolvimento de sistemas conectados em rede ressaltou a importância dos conhecimentos adquiridos nas disciplinas do curso de Graduação em Engenharia Elétrica. O estágio proporcionou contato com situações e pessoas de diferentes segmentos da empresa, que me agregaram experiência e vários momentos de aprendizagem de grande valia para um profissional.

Referências

- [1] Mattern, F. and Floerkemeier, C. (2010). From the internet of computers to the internet of things. In From active data management to event-based systems and more, pages 242–259. Springer.
- [2] Harris, D. and Harris, S. Digital Design and Computer Architecture. [S.l.]: Elsevier Science, 2012. ISBN 9780123978165.
- [3] <http://www-cs-faculty.stanford.edu/~uno/abcde.html>. Acessado em 09/11/2018.
- [4] <http://www.c2o.pro.br/automacao/ar01s11.html>. Acessado em 09/11/2018.
- [5] https://secure.php.net/manual/pt_BR/intro-what-is.php. Acessado em 09/11/2018.
- [6] <http://www.ti.com/lit/ds/symlink/lm1117.pdf>. Acessado em 09/11/2018.
- [7] www.ti.com/lit/ds/symlink/max232.pdf. Acessado em 09/11/2018.