

Faculdade de Ciência e Tecnologia de Montes Claros

Felipe Israel Corrêa

**RELATÓRIO DE ESTÁGIO SUPERVISIONADO EM
ENGENHARIA DE COMPUTAÇÃO**

Montes Claros - MG
2017

Felipe Israel Corrêa

**RELATÓRIO DE ESTÁGIO SUPERVISIONADO EM ENGENHARIA
DE COMPUTAÇÃO**

**Relatório de Estágio Supervisionado
apresentado ao Curso de Engenharia de
Computação, da Faculdade de Ciência e
Tecnologia de Montes Claros como parte dos
requisitos para a obtenção do título de
Engenheiro de Computação.**

**Empresa: Faculdade de Ciências e Tecnologia
de Montes Claros.**

**Supervisor: MAURÍLIO JOSÉ INÁCIO,
Coordenador do Curso de Engenharia da
Computação.**

**Montes Claros - MG
2017**

RESUMO

Este relatório tem como objetivo apresentar a importância do estágio supervisionado e as principais atividades desenvolvidas pelo acadêmico no laboratório de Sistemas Computacionais da Faculdade de Ciência e Tecnologia de Montes Claros (FACIT). No primeiro capítulo será feita uma breve apresentação da empresa seguido pelos capítulos que abrangem as atividades desenvolvidas durante o estágio, suas metodologias, resultados e conclusão. Através deste documento fica exposta a importância da ligação entre os estudos práticos e teóricos, vistos durante todo o período de graduação, para a formação do engenheiro.

Palavras chaves: Estágio, Sistemas Computacionais, formação.

LISTA DE FIGURAS

Figura 1 – Raspberry Pi 1 Model B	9
Figura 2 – Fluxo de desenvolvimento dos experimentos	10
Figura 3 – Diagrama simplificado do circuito cubo	11
Figura 4 – Diagrama do circuito cubo	12
Figura 5 – Circuito cubo montado na protoboard.....	12
Figura 6 – Interface gráfica do aplicativo cubo	13
Figura 7 – Diagrama do circuito relé	15
Figura 8 – Circuito relé montado na protoboard	15
Figura 9 – Interface gráfica do aplicativo relé	16
Figura 10 – Diagrama do circuito relé web	17
Figura 11 – Código fonte utilizando protocolo CGI	18
Figura 12 – Interface web projetada	18
Figura 13 – Servidor web em funcionamento	19
Figura 14 – Diagrama do circuito LCD	21
Figura 15 – Tela de escolha exibida para o usuário	22
Figura 16 – Exibição da temperatura	22

SUMÁRIO

INTRODUÇÃO.....	5
CAPÍTULO 1 APRESENTAÇÃO DA EMPRESA	6
CAPÍTULO 2 ATIVIDADES DESENVOLVIDAS.....	8
2.1 Organização do laboratório e auxílio aos estudantes que iam visitá-lo	8
2.2 Raspberry Pi	8
2.3 Experimento 1 - Controle de um cubo de led	10
2.4 Experimento 2 - Controle de um módulo de relés via interface gráfica	12
2.5 Experimento 3 - Controle de um módulo de relés via interface web	14
2.6 Experimento 4 - Controle e visualização de sensores via display LCD	16
CAPÍTULO 3 ARTICULAÇÃO ENTRE A PRÁTICA DE ESTÁGIO E A TEORIA ACADÊMICA.....	23
4 CONCLUSÃO.....	20
REFERÊNCIAS BIBLIOGRÁFICAS.....	25

INTRODUÇÃO

O estágio supervisionado é de extrema importância para a formação e capacitação do acadêmico, uma vez que o prepara para mercado de trabalho, que hoje, está cada vez mais exigente.

A disciplina de estágio supervisionado faz parte da grade curricular do curso de Engenharia de Computação da Faculdade de Ciência e Tecnologia de Montes Claros – FACIT, e, através dele, o graduando tem a possibilidade de unir os conhecimentos teóricos aos práticos, possibilitando resolução de problemas emergentes durante a atuação laboral, com diversos graus de dificuldades. Tem-se aí um crescimento pessoal e profissional.

O presente documento fornece informações detalhadas sobre as atividades desenvolvidas, no laboratório de Sistemas Computacionais, por um acadêmico do curso de Engenharia da Computação, visando listar a importância do estágio para sua formação.

CAPÍTULO 1 APRESENTAÇÃO DA EMPRESA

Em 1976 foi criada a Fundação Educacional de Montes Claros (FEMC), pela Associação Comercial e Industrial de Montes Claros. Nascia com o objetivo de fornecer e qualificar profissionais operacionais e técnicos para a implantação do Parque Industrial Norte Mineiro.

É uma entidade de direito privado, sem fins lucrativos, dirigida por empresários representantes das empresas mantenedoras e à sua criação instalou a Escola Técnica, com os cursos de Comercialização e Mercadologia, Eletrônica, Eletromecânica, Edificações e Segurança do Trabalho.

Foi certificada pela OIT e UNESCO após auditoria realizada em 1990, onde por estas empresas passou a ser divulgada mundialmente com relação a qualidade de trabalho em educação profissionalizante e o modo “sui generis” de financiamento da educação de carentes por parte das empresas. Em 2001 foram investidos mais de R\$ 3.500.000,00 na reestruturação e expansão da estrutura física e laboratorial da Escola Técnica, graças ao empenho da diretoria presidida pelo Dr. Ariovaldo de Melo Filho.

Considerada pelo Ministério da Educação e Cultura (MEC) em 2002 como modelo, o projeto pedagógico da Escola Técnica foi encaminhado para ONU como referência nacional em Educação para o Trabalho.

Atualmente a Escola Técnica possui cerca de 800 alunos matriculados.

Em dezembro de 2000 a FEMC, em parceria com o SEBRAE, inaugura uma incubadora para empresas de base tecnológica, chamada INCET e já recebeu, em convênio com o MEC investimentos na ordem de R\$ 108.040,00.

A FEMC também mantém em seu portfólio o Colégio Delta, que atua no ensino médio e consta com cerca de 450 alunos.

Além da Escola Técnica e do Colégio Delta, a FEMC criou em agosto de 2002 a Faculdade de Ciência e Tecnologia (FACIT), inaugurada com os cursos de Engenharia da Computação, Engenharia de Controle e Automação, Engenharia Química e Engenharia de Telecomunicações.

Foi concebida com a missão de contribuir para o desenvolvimento integral do ser humano e da sociedade, visando transformar-se num centro de excelência e referência em educação, pesquisa e prestação de serviço mantendo os valores de amor, ética, cidadania, respeito, responsabilidade, confiança e proatividade.

A FACIT surge pela falta de cursos superiores na área de tecnologia em todo norte de Minas, vale do Jequitinhonha e sul da Bahia. Atualmente, além dos cursos criados da

inauguração, conta com os cursos de Engenharia Civil, Engenharia de Produção, Engenharia Elétrica e Engenharia Mecânica, além dos cursos de Pós-graduação em Engenharia de Manutenção e Pós-graduação em Gestão da Qualidade.

Criados dentro dos princípios e filosofia da FEMC, os cursos superiores da FACIT objetivam desenvolver competências, habilidades e atitudes em um cidadão, preparando-o para conquistar sua viabilização econômica, tecnológica com consequente melhoria da qualidade de vida e distribuição de renda para a população das regiões norte e nordeste de Minas.

A FACIT proporciona aos seus alunos uma formação de qualidade para que este possa responder às exigências do mercado de trabalho, através de docentes qualificados e diversos laboratórios com equipamentos e materiais que refletem à realidade do mundo atual.

A instituição ainda possui um projeto na área social, sem fins lucrativos, chamado Projeto Social Juventude Cidadã, objetivando repassar para os jovens que participam, assuntos relacionados com a educação, cultura e tecnologia. Através deste projeto busca-se o crescimento e o desenvolvimento pleno do jovem, principalmente no aspecto social.

CAPÍTULO 2 ATIVIDADES DESENVOLVIDAS

Em todo o período de estágio várias atividades foram desenvolvidas pelo acadêmico, a fim de proporcionar o aperfeiçoamento dos conhecimentos e colocar em prática toda a teoria vista em períodos anteriores. O estágio realizado teve como base o desenvolvimento de experimentos utilizando a placa embarcada Raspberry Pi.

Todos os diagramas apresentados neste relatório foram feitos utilizando o software gratuito para modelagem de circuitos Fritzing.

Neste capítulo serão descritos, detalhadamente, os experimentos desenvolvidos, a metodologia empregada e os elementos utilizados para a realização dos mesmos. Foram desenvolvidos no decorrer do estágio quatro experimentos:

- Experimento 1 - Controle de um cubo de led;
- Experimento 2 - Controle de um módulo de relés via interface gráfica;
- Experimento 3 - Controle de um módulo de relés via interface web;
- Experimento 4 - Controle e visualização de sensores via display LCD;

2.1 Organização do laboratório e auxílio aos estudantes que iam visitá-lo

No período de estágio o acadêmico ficou responsável por manter o laboratório organizado, além de auxiliar os estudantes que iam ao laboratório desenvolver práticas, o que proporcionou um maior contato com os equipamentos e componentes de eletrônica, tais como led's, resistores, capacitores, transistores, etc.

Foi possível também uma interação e aperfeiçoamento no conhecimento das placas de sistemas embarcados, tais como Arduino, Raspberry Pi, MSP430, com os seus cabos e shields para aplicações específicas.

2.2 Raspberry Pi

Tendo em vista que a base do estágio foi o desenvolvimento de aplicações em Raspberry Pi, neste capítulo iremos detalhar seu histórico e funcionamento.

A Raspberry Pi é um computador do tamanho de um cartão de crédito, que pode ser plugado a qualquer monitor e é capaz de executar tarefas que um computador desktop faz.

A ideia da Raspberry Pi surgiu pelas mentes de Eben Upton, Rob Mullins, Jack Lang e Alan Mycroft em 2006 na Universidade de Cambridge, com o intuito de criar um computador pequeno e barato para crianças.

Em 2008 é fundada a Raspberry Pi Foundation com o objetivo de desenvolver a placa, o que foi facilitado pela diminuição dos custos e aumento da capacidade de processadores para dispositivos móveis.

Os primeiros modelos foram desenvolvidos com o processador Atmel ATmega644, mas o alto custo dos demais componentes e a baixa eficiência, acabaram por ir contra os princípios da Raspberry Pi Foundation.

Após anos de pesquisa em 2011 é lançado o protótipo com processadores ARM, que são utilizados desde então.

Nos experimentos foi utilizada a placa Raspberry Pi 1 Model B, lançada em 2012 com os seguintes componentes:

- Processador ARM1176JZF-S;
- 700MHz de clock;
- Tecnologia de 32 bits;
- 512 MB de memória RAM;
- Duas portas USB 2.0;
- Porta Ethernet;
- Conector de vídeo HDMI;
- Slot para cartão SD;

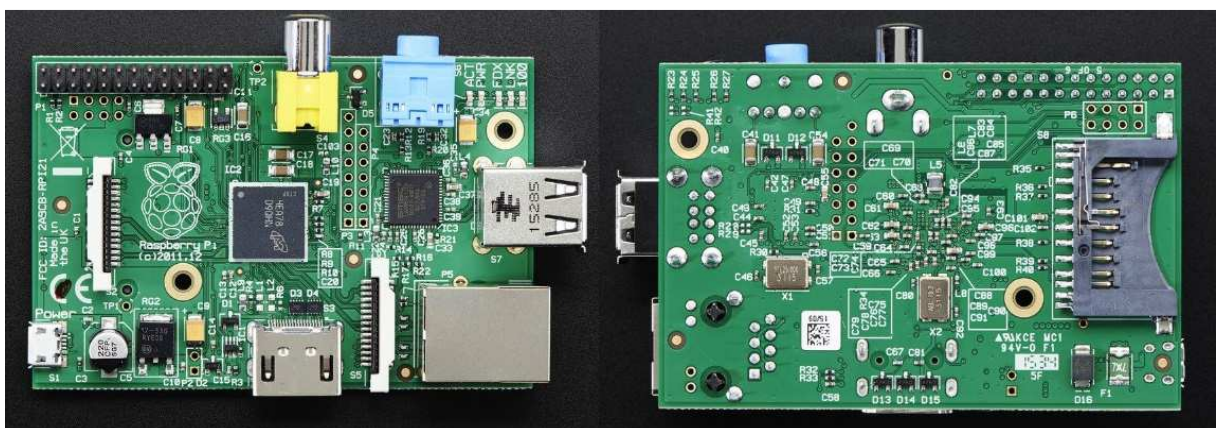


Figura 1: Raspberry Pi 1 Model B

Fonte: Adafruit (<https://www.adafruit.com/product/998>)

O fluxo de desenvolvimento dos experimentos seguiu a lógica apresentada na figura 2:

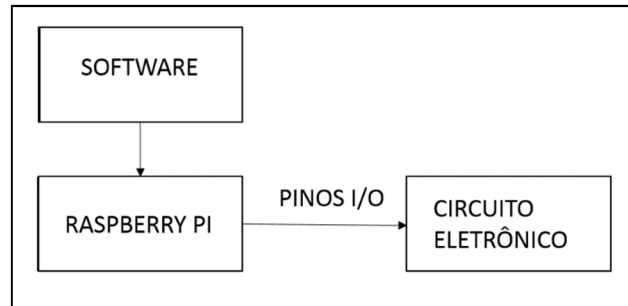


Figura 2: Fluxo de desenvolvimento dos experimentos
Fonte: O Autor

Os softwares foram desenvolvidos em linguagem Python 3.5, na IDE disponível no sistema operacional da Raspberry Pi, chamado Raspbian, e através da sua execução os pinos de I/O são configurados para interferir no circuito eletrônico.

2.3 Experimento 1 - Controle de um cubo de led

O objetivo deste experimento consistiu em controlar um cubo de led 3x3x3, possibilitando que as sequências fossem alteradas de acordo com o desejo do usuário.

Para o projeto foram utilizados os seguintes componentes:

- **27 led's de 5mm verdes:** criam os efeitos definidos pelo usuário;
- **9 resistores de 180Ω:** limitam o fluxo de corrente em cada ânodo do led;
- **3 resistores de 10KΩ:** limitam o fluxo de corrente nas bases dos transistores;
- **3 transistores BC548 NPN:** responsáveis por acionar ou desligar determinada coluna ou linha do cubo, criando o efeito determinado pelo usuário;

2.3.1 Funcionamento

Este cubo de led foi desenvolvido para possuir três níveis com 9 led's. Em cada coluna os cátodos dos led's são conectados de três em três e ligados diretamente a um dos pinos de I/O da Raspberry.

Já os ânodos pertencentes a uma mesma linha são conectados comumente ao emissor de um dos transistores, conforme visto de forma simplificada na figura 3:

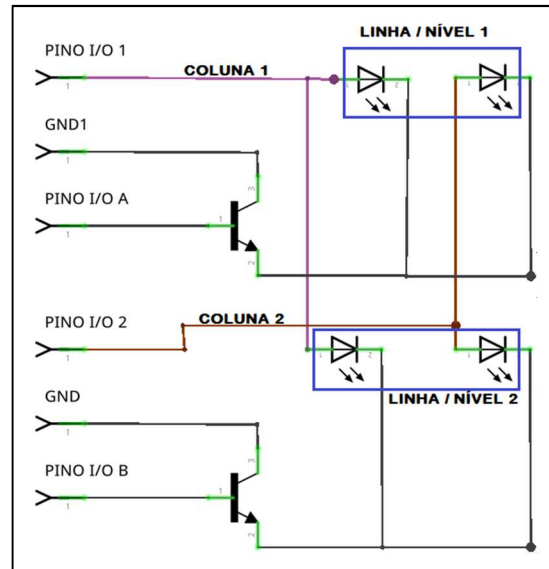


Figura 3: Diagrama simplificado do circuito cubo
Fonte: O Autor

O acendimento de um determinado led ou conjunto de led's vai depender da coluna e linha energizada.

O transistor neste circuito trabalha como um a chave liga/desliga. Ao aplicar um nível alto em sua base, o valor da corrente que está no coletor, que neste caso está conectado ao GND da Raspberry, irá fluir para o emissor, fazendo com que os ânodos recebam nível baixo. Possibilitando assim, ocorrer o acionamento do led ou conjunto de led's que estiverem setados em nível alto.

- `interface.py`: possui os comandos e métodos para geração da interface gráfica;
- `led_cube.py`: possui os métodos para configuração dos pinos de I/O da Raspberry e os métodos de animação do cubo;

- `main.py`: arquivo principal que possibilita a conexão entre os métodos de geração da interface gráfica e seus componentes com os métodos de animação do cubo;

A figura 6 apresenta a interface do programa criado:

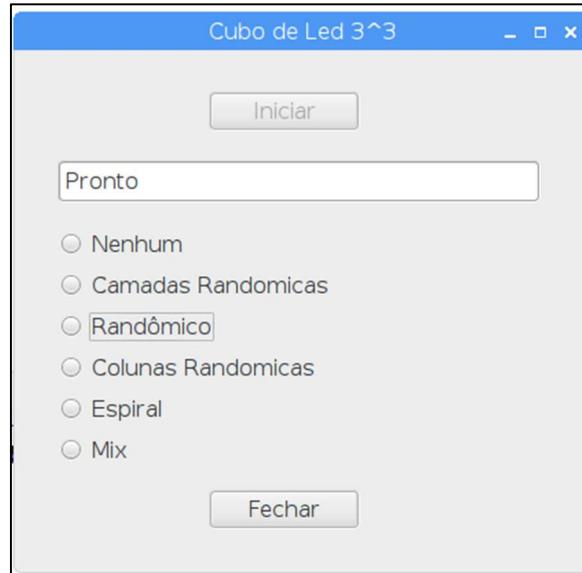


Figura 6: Interface gráfica do aplicativo cubo
Fonte: O Autor

Através deste experimento foi possível aplicar os conhecimentos de eletrônica pois foi necessária a montagem do circuito, bem como os conhecimentos de programação para criação da interface gráfica.

A prática possibilitou também aprofundar um pouco mais o conhecimento sobre a placa Raspberry Pi 1.

2.4 Experimento 2 - Controle de um módulo de relés via interface gráfica

Este experimento teve como objetivo controlar um módulo relé de quatro canais para acender e apagar lâmpadas.

Para o desenvolvimento da prática foram utilizados os seguintes componentes:

- **4 led's azuis:** foram utilizados para simular o funcionamento de lâmpadas;
- **4 resistores 330Ω:** para controlar o fluxo de corrente nos ânodos de cada led;
- **1 módulo relé com 4 canais:** responsável por gerar os sinais de ligar ou desligar a lâmpada;

2.4.1 Funcionamento

O módulo relé utilizado neste experimento possui 8 entradas. As entradas VCC e GND são utilizadas para alimentar o módulo com a tensão fornecida pela placa de desenvolvimento. Caso seja necessário o uso de uma fonte externa de alimentação utiliza-se as entradas JD-VCC e VCC.

Neste projeto a fonte de tensão foi a própria Raspberry, portanto utilizou-se as entradas VCC e GND e as entradas JD-VCC e VCC foram curto-circuitadas, conforme determina o fabricante.

O módulo possui ainda as entradas IN1, IN2, IN3 e IN4 referentes a cada relé e ao serem setadas, tanto em nível alto como baixo, efetuam a comutação dos relés. Cada uma das entradas IN foi conectada a um pino I/O da Raspberry.

No módulo ainda existem quatro led's que indicam o status de cada relé, ficando acesos caso ele esteja com sua entrada setada em nível alto e apagado caso contrário.

Os relés possuem três contatos NO (normalmente aberto), NC (normalmente fechado) e COM (comum ou central).

O contato NO são os que estão abertos enquanto a bobina não está energizada e se fecham quando a mesma recebe corrente. O contato NC funciona de forma contrária ao contato NO. Já o contato COM estabelece a condução de corrente com o contato que estiver fechado.

Através deste entendimento o circuito foi montado da seguinte maneira:

- O contato COM de cada relé foi ligado a alimentação de 3,3V da Raspberry;
- Os resistores de 330ohms foram ligados em série com os ânodos de cada led no contato NO;
- Os cátodos dos led's foram conectados ao pino de GND da Raspberry, conforme mostrar a figura 7:

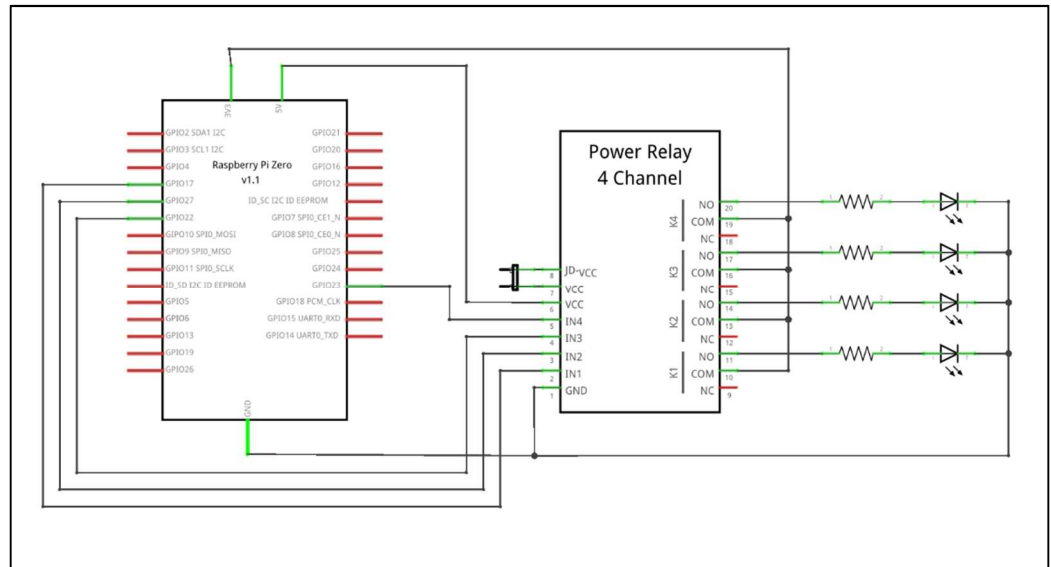


Figura 7: Diagrama do circuito relé

Fonte: O Autor

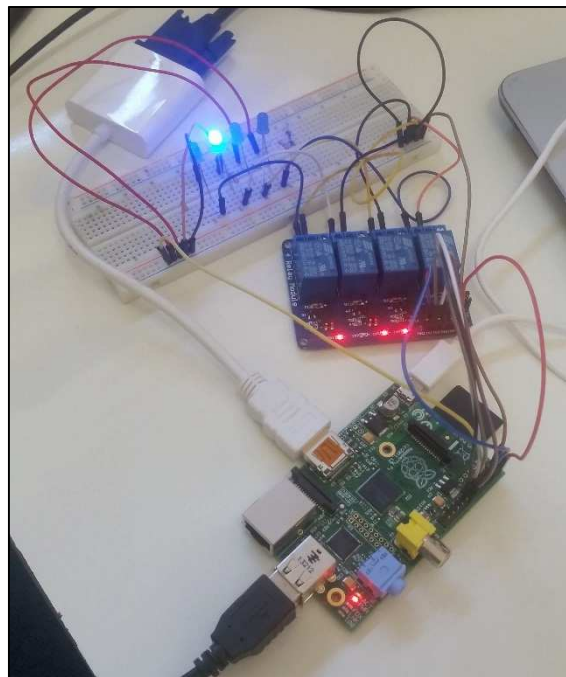


Figura 8: Circuito relé montado na protoboard

Fonte: O Autor

A interface gráfica, também foi desenvolvida em PyQt e foi projetada para que o usuário pudesse escolher entre acionar cada lâmpada individualmente ou em conjunto. Para atender a estes objetivos foi necessária a criação de três arquivos contendo o código de programação, que são:

- interface.py: possui os comandos e métodos para geração da interface gráfica;

- rele.py: os métodos para configuração dos pinos de I/O da Raspberry e para acionamento dos relés;
- main.py: arquivo principal que possibilita a conexão entre os métodos de geração da interface gráfica e seus componentes com os métodos de acionamento dos relés;



Figura 9: Interface gráfica do aplicativo relé
Fonte: O Autor

Com esta prática além do aprofundamento em eletrônica e programação, foi observado e consolidado o conhecimento sobre o processo que a placa Raspberry Pi 1 faz em tratar os sinais individual ou coletivamente.

2.5 Experimento 3 - Controle de um módulo de relés via interface web

Este experimento objetivou controlar um módulo relé com quatro canais via interface web. Para tal, a placa Raspberry Pi atuou como um servidor de aplicação web.

Para este experimento foi necessário somente o uso de um módulo relé de quatro canais e seu funcionamento foi explicado no experimento 2.

Na figura 10, pode-se ver como o circuito foi projetado:

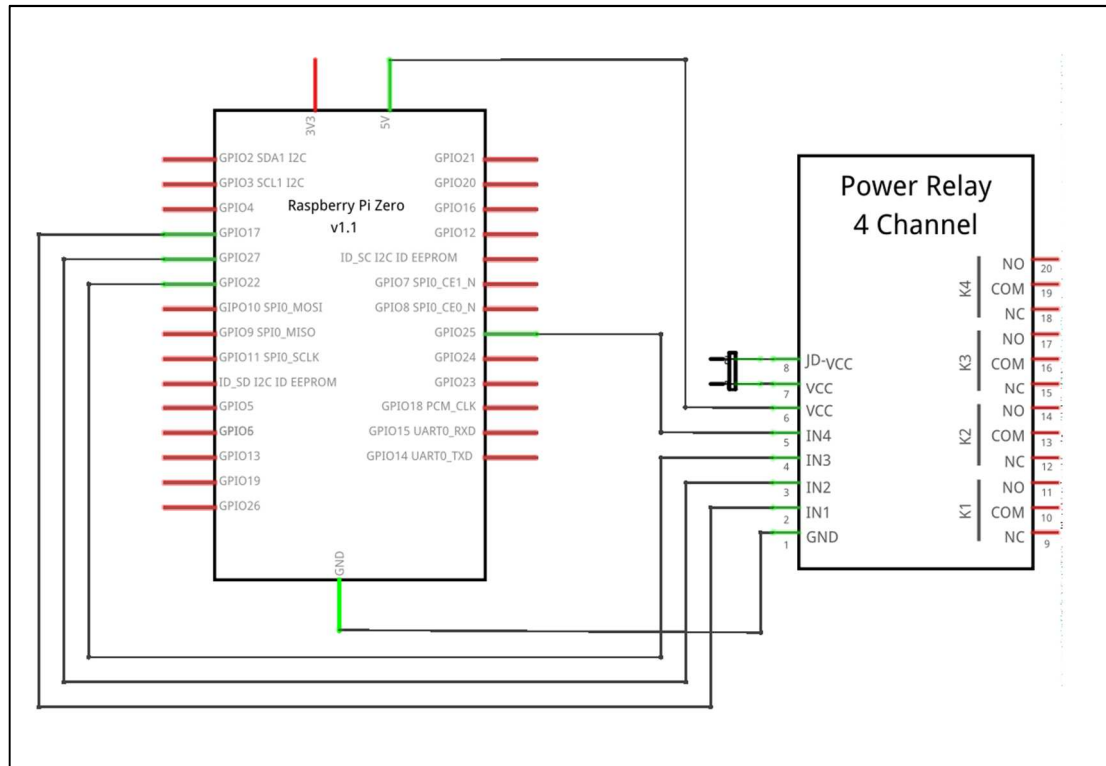


Figura 10: Diagrama do circuito relé web
Fonte: O Autor

2.5.1 Funcionamento

A interface gráfica foi desenvolvida através do protocolo chamado Common Gateway Interface (CGI), que fornece integração entre os servidores HTTP e as aplicações Web.

Este protocolo possibilita que o servidor web informe uma série de parâmetros para o programa que deve entregar uma resposta para este mesmo servidor, não importando a linguagem utilizada. Portanto todo o código pôde ser escrito em linguagem Python, informando os parâmetros como linguagem HTML, conforme visto na figura 11:

```

1  #!/usr/bin/python3.5
2
3  import cgi
4  import cgitb
5
6  cgitb.enable()
7  form = cgi.FieldStorage()
8
9  print("Content-type:text/html")
10 print()
11 print("<html>")
12 print("<head>")
13 print("<title>Acionamento Reles</title>")
14 print("<center><h1>Controle de relés</h1></center><br><br>")
15 print("<style> h1{color:blue; size:px;}</style>")
16 print("</head>")
17 print("<body>")
18 print("<center>")
19 print("<h2>Controle</2>")
20 print("</center>")
21 print("</body>")
22 print("</html>")

```

Figura 11: Código fonte utilizando protocolo CGI

Fonte: O Autor

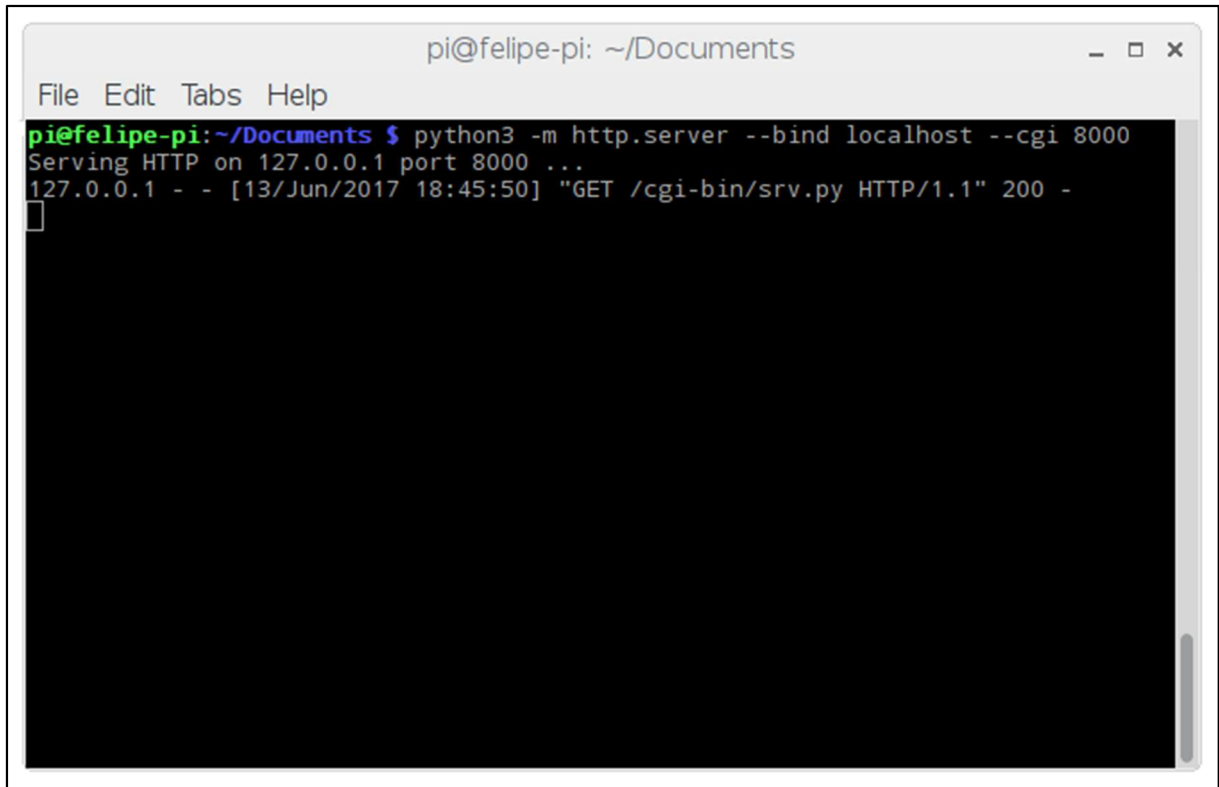
Esta interface gráfica foi projetada para que o usuário pudesse acionar individualmente os relés. Foi necessária a criação de dois arquivos contendo o código de programação, que são:

- rele.py: contém os métodos para configuração dos pinos de I/O da Raspberry e acionamento dos relés;
- server.py: contém os comandos e métodos para geração do protocolo CGI, da página em HTML e faz a integração com os métodos de acionamento dos relés;



Figura 12: Interface web projetada

Fonte: O Autor

A screenshot of a terminal window titled 'pi@felipe-pi: ~/Documents'. The window has a menu bar with 'File', 'Edit', 'Tabs', and 'Help'. The terminal shows the command 'python3 -m http.server --bind localhost --cgi 8000' being executed. The output indicates the server is running on 127.0.0.1 port 8000. A log entry shows a GET request to /cgi-bin/srv.py with a 200 status code. The terminal background is black with green and white text.

```
pi@felipe-pi: ~/Documents
File Edit Tabs Help
pi@felipe-pi:~/Documents $ python3 -m http.server --bind localhost --cgi 8000
Serving HTTP on 127.0.0.1 port 8000 ...
127.0.0.1 - - [13/Jun/2017 18:45:50] "GET /cgi-bin/srv.py HTTP/1.1" 200 -
```

Figura 13: Servidor web em funcionamento

Fonte: O Autor

Através desta prática pôde-se observar e entender os processos que fazem a placa Raspberry Pi atuar como um servidor web, além de aprofundar os conhecimentos adquiridos nas matérias relacionadas com rede de computadores.

2.6 Experimento 4 - Controle e visualização de sensores via display LCD

O objetivo deste experimento consistiu em poder selecionar a informação desejada, possibilitando que esta fosse exibida via display de LCD.

A seleção da informação foi controlada por dois botões (push-button).

Os materiais necessários para este projeto foram:

- **2 sensores infravermelho:** utilizados para detecção de presença;
- **1 sensor DHT11:** para temperatura e umidade;
- **2 push-button:** para controle das opções;
- **1 resistor de 4,7K Ω :** utilizado entre os pinos VCC e DATA do sensor DHT11 para controlar o fluxo de corrente;
- **1 display LCD 16x2:** responsável por exibir as informações;
- **1 potenciômetro de 10K Ω :** utilizado para ajustar o contraste do display;

2.6.1 Funcionamento

O sensor DHT11 utilizado neste experimento possui quatro pinos, sendo o pino 1 de alimentação VCC, o pino 2 de dados (sinal) e o pino 4 o GND. O pino 3 não é funcional.

Possui internamente dois tipos de sensores, um tipo termistor NTC para temperatura e um tipo HR202 para umidade. Permite medir temperaturas de 0 a 50 graus Célsius com erro de 2 graus e umidade relativa na faixa de 20 a 90% com erro de até 5%.

As leituras são enviadas pelo mesmo barramento através do formato dos dados: 8 bits inteiro Umidade + 8 bits decimal Umidade + 8 bits inteiro Temperatura + 8 bits decimal Temperatura + 8 bits check sum (verificação de soma) = 40 bits. Caso o valor do check sum seja diferente do total dos valores de umidade e temperatura o sensor reconhece a leitura como um erro. No código usa-se uma biblioteca específica para tratamento destes dados.

Os sensores de presença utilizados são do tipo infravermelho ou piroelétricos contendo 3 pinos, sendo o pino 1 alimentação VCC, pino 2 de dados (sinal) e o pino 3 GND.

Possuem em seu interior um material cristalino que gera uma corrente elétrica em sua superfície quando expostos ao calor sob a forma de radiação infravermelha. Esta situação faz com que um transistor FET interno seja acionado e emita um sinal de nível alto no pino de dados. Caso não haja detecção de movimento o sinal enviado será de nível baixo.

É possível ainda ajustar a duração do tempo de espera da estabilização, ou seja, após acionado ele permanecerá assim um determinado período de tempo mesmo sem detectar outro movimento e ajustar sua sensibilidade. Estes ajustes são feitos nos potenciômetros disponíveis no corpo do sensor.

O display LCD que foi utilizado é do tipo 16x2, portanto 16 linhas e 2 colunas. Sua comunicação com a placa é feita através dos pinos DB4 a DB7 que recebem 1 byte dividido em 2 remessas de 4 bits com a configuração ou caractere a ser escrito.

Os demais pinos que são conectados a placa são:

- 1 – V_{ss} (GND);
- 2 - V_{dd} (5V);
- 4 – RS (Register select), definir: 1 – Dado ou 0 – Instrução;
- 5 – R/W, definir: 1 – Leitura ou 0 – Escrita;
- 6 – E, definir: 1 – Habilitado ou 0 – Desabilitado;
- 15 e 16 – ânodo e cátodo para contraste, respectivamente;

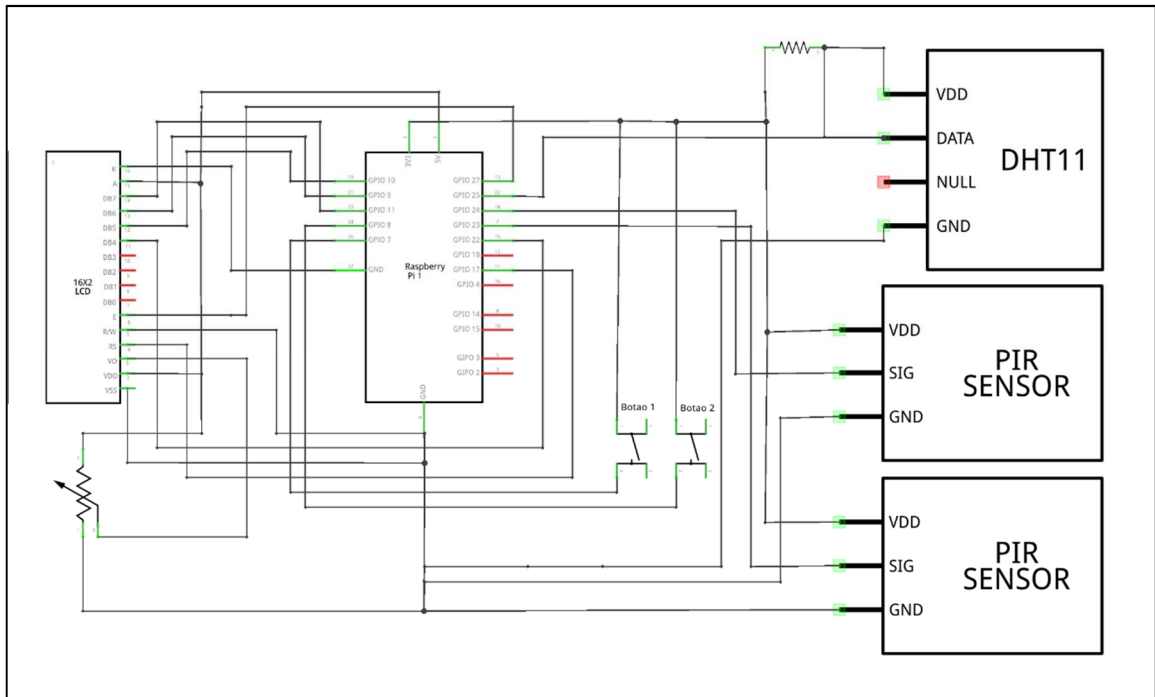


Figura 14: Diagrama do circuito LCD

Fonte: O Autor

Ao usuário era possível verificar as informações sobre a temperatura, umidade e os estados dos sensores de presença, se ativos, caso tivessem detectado alguma presença ou se desativados, ou seja, sem detectar presença. A figura 15 exemplifica a possibilidade do usuário em escolher entre o sensor ATMOSFÉRICO (temperatura e umidade) ou de PRESENÇA.

A programação do projeto foi feita em linguagem Python, através de um único arquivo, chamado lcd.py, que contém todos os métodos e comandos para configuração dos pinos da Raspberry, botões e exibição das informações, além das bibliotecas necessárias para o funcionamento do display LCD e do sensor DHT11.

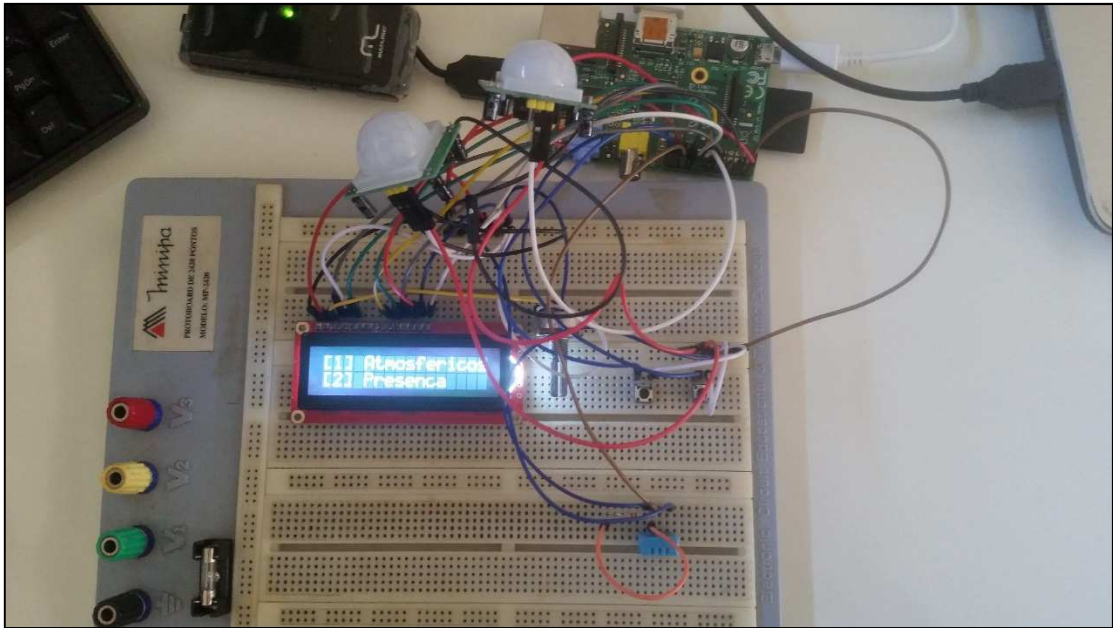


Figura 15: Tela de escolha exibida para o usuário
Fonte: O Autor

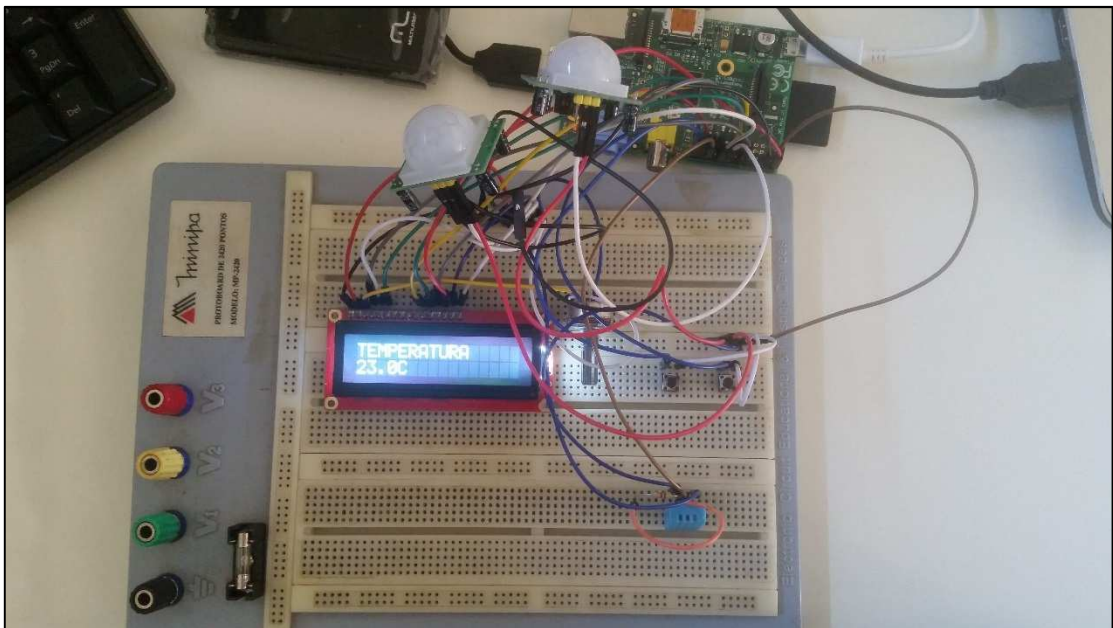


Figura 16: Exibição da temperatura
Fonte: O Autor

Nesta prática além do aprofundamento em eletrônica, foi possível observar e entender os processos utilizados pela placa Raspberry Pi para receber e tratar os sinais de entrada enviados pelos sensores.

CAPÍTULO 3 ARTICULAÇÃO ENTRE A PRÁTICA DE ESTÁGIO E A TEORIA ACADÊMICA

As atividades executadas pelo acadêmico foram baseadas nas disciplinas vistas durante o período de graduação, ajudando a reforçar ainda mais o conteúdo destes assuntos. Como foco o estágio teve disciplinas como Eletrônica, Sistemas Digitais e Microprocessados além das matérias relacionadas à programação de computadores, ou seja, Algoritmo.

A disciplina de eletrônica fornece ao acadêmico as bases para leis e ferramentas relacionadas aos circuitos compostos por elementos, tais como resistores, capacitores, fontes de tensão, etc.

Já as disciplinas de sistemas digitais e microprocessados tratam de assuntos relacionados a funções lógicas, álgebra booleana, tecnologias, funcionamento e projeto de processadores, funcionamento de sistemas microcontrolados, dentre outros.

Em relação as matérias de algoritmo, ao acadêmico são apresentados temas relacionados à lógica de programação, estrutura de dados e arquivos, programação orientada a objetos, etc.

Todo este conhecimento adquirido nas aulas teóricas foi de grande importância para a realização do estágio e desenvolvimento das práticas descritas.

4 CONCLUSÃO

O estágio possibilita ao participante aplicar, na prática, todo conteúdo teórico visto durante todo o período de graduação, portanto, esta etapa é de suma importância para o acadêmico.

Durante todo este período de estágio, o acadêmico teve contato com inúmeros equipamentos e procedimentos que farão parte do seu cotidiano profissional, aumentando sua capacidade de crítica e fixando seu conhecimento nos assuntos tratados.

O estágio propiciou assim um maior conhecimento prático e teórico nos assuntos relacionados a eletrônica, programação de computadores, pesquisa e elaboração de projetos.

REFERÊNCIAS BIBLIOGRÁFICAS

FACIT. **Estágio – Manual para normalização da redação v2017**. Disponível em: <<http://www.femc.edu.br/uploads/arquivos/Estágio - Manual para normalização da redação v2017.pdf>>. Acesso em 10 de junho 2017.

FEMC. **Histórico**. Disponível em: <<http://www.femc.edu.br/portal/femc/pagina/49-historico>>. Acesso em 13 de junho de 2017.

Adafruit. **Raspberry Pi Model B**. Disponível em: <<https://www.adafruit.com/product/998>>. Acesso em 14 de junho de 2017.

Makersify. **The History of the Raspberry Pi**. Disponível em: <<https://makersify.com/blogs/makersify-blog/the-history-of-the-raspberry-pi>>. Acesso em 14 de junho de 2017.

Elinux. **RPi General History**. Disponível em: <http://elinux.org/RPi_General_History>. Acesso em 14 de junho de 2017.

Embarcados. **Módulo de display LCD**. Disponível em: <<https://www.embarcados.com.br/modulo-de-display-lcd/>> . Acesso em 25 de junho de 2017.

Infoescola. **Relê**. Disponível em: <<http://www.infoescola.com/electronica/rele/>>. Acesso em 25 de junho de 2017.

Micropik. **DHT11 Humidity & Temperature Sensor**. Disponível em: <<http://www.micropik.com/PDF/dht11.pdf>>. Acesso em 26 de junho de 2017.

Nova Eletrônica. **Sensor Infravermelho Passivo**. Disponível em: <<http://blog.novaeletronica.com.br/sensor-pir-sensor-infravermelho-passivo/>>. Acesso em 26 de junho de 2017.