



**FACULDADE DE TECNOLOGIA SENAI “ANCHIETA”
CURSO SUPERIOR DE TECNOLOGIA EM ELETRÔNICA INDUSTRIAL**

**DAVI MONTE DE OLIVEIRA
FABRINI SILVA GOMES
LEONARDO RODRIGUES TORRES
JOEL PINHO BASTOS**

CARGA ELETRÔNICA CONTROLADA PARA CORRENTE CONTÍNUA

**SÃO PAULO
2020**

**DAVI MONTE DE OLIVEIRA
FABRINI SILVA GOMES
LEONARDO RODRIGUES TORRES
JOEL PINHO BASTOS**

CARGA ELETRÔNICA CONTROLADA PARA CORRENTE CONTÍNUA

Trabalho de Conclusão de Curso apresentado a Faculdade de Tecnologia SENAI “Anchieta” como requisito para obtenção do título de Tecnólogo em Eletrônica Industrial.

Orientadores: Professor Vander Nunes; Professor

**SÃO PAULO
2020**

Leonardo Torres
Joel Pinho
Davi Monte
Fabrini Silva Gomes

CARGA ELETRÔNICA CONTROLADA PARA CORRENTE CONTÍNUA

Trabalho de Conclusão de Curso apresentado a Banca Examinadora como requisito para obtenção do título de Tecnólogo em Eletrônica Industrial.

Nome do aluno

Orientador

Prof. Dionny Cleverson Mazio Batista

Orientador

Prof. Vander Celio Nunes

Banca examinadora

Prof. Erineu Claudemir

Prof. Fernando Faria

São Paulo, _____ de _____ 2020.

Agradecimentos

Primeiramente, agradecemos a Deus acima de tudo, aos nossos familiares, amigos, e docentes pelo apoio, sem vocês, não chegaríamos a este grande dia.

Resumo

O presente trabalho aqui apresentado representa o projeto de um equipamento chamado de Carga Eletrônica para corrente contínua. A carga eletrônica consiste de um equipamento destinado a realizar testes em aparelhos geradores de energia como fontes de alimentação, baterias, carregadores de celular, módulos fotovoltaicos, conversores CC, entre outros, sendo assim um instrumento de grande utilidade em laboratórios de desenvolvimento, salas de aulas técnicas e linhas de produção industrial. O projeto aqui apresentado foi desenvolvido em sala de aula sob orientação dos professores de PEA e através de pesquisas de patentes de diversos equipamentos similares através das quais foi possível obter informações pertinentes e importantes para a implementação dos circuitos elétricos envolvidos neste projeto. Uma vez definidos os circuitos elétricos estes foram testados em ambiente virtual pelo software Proteus ISIS onde pode-se comprovar o seu funcionamento. Utilizou-se também o software CCS C Compiler onde através do qual foi possível desenvolver em linguagem C, todo o código do programa para gerenciamento e controle do projeto em questão. Após as simulações e acertos necessários para o correto funcionamento do projeto, foram desenvolvidas as placas de circuito impresso no software Proteus ARES e estas confeccionadas através da técnica de transferência térmica. O projeto Carga Eletrônica CC foi então montado e testado, atuando com carga de uma fonte de alimentação chaveada de 12,0V / 10A onde os resultados obtidos demonstraram a eficácia e excelente desempenho do projeto aqui apresentado.

Palavras-chave: Carga Eletrônica, Corrente Contínua, PWM, Carga Dinâmica.

Abstract

The work presented here represents the design of an equipment called Electronic Load for direct current. The electronic charge consists of equipment designed to carry out tests on energy generating devices such as power supplies, batteries, cell phone chargers, photovoltaic modules, DC converters, among others, thus being a very useful instrument in development laboratories, technical classes and industrial production lines. The project presented here was developed in the classroom under the guidance of PEA teachers and through patent searches for various similar equipment through which it was possible to obtain relevant and important information for the implementation of the electrical circuits involved in this project. Once the electrical circuits were defined, they were tested in a virtual environment by the Proteus ISIS software, where they can be checked for operation. The CCS C Compiler software was also used, through which it was possible to develop in C language, all the program code for management and control of the project in question. After the simulations and adjustments necessary for the correct functioning of the project, the printed circuit boards were developed in the Proteus ARES software and made using the thermal transfer technique. The DC Electronic Load project was then assembled and tested, operating under load from a 12.0V / 10A switched power supply where the results obtained demonstrated the efficiency and excellent performance of the project presented here.

Key-words: Eletronic Charge, Direct Current , PWM, Dinamic Charge.

Lista de Figuras

Este item é opcional, porém recomendado quando houver pelo menos 3 figuras/fotos/diagramas ao longo do texto.

Figura 1 - Célula de Potência	16
Figura 2 - Módulo de Potência	16
Figura 3 - Diagrama Esquemático Módulo de Potência	17
Figura 4 - Mosfet IRFP264	17
Figura 5 - Amplificador Operacional LM 358	18
Figura 6 - Circuito de Controle e gerenciamento	19
Figura 7 - Diagrama Elétrico PIC18F4550	20
Figura 8 - Diagrama Esquemático do Teclado	21
Figura 9 - Teclado no Painel Frontal	21
Figura 10 - Display	22
Figura 11 - Amplificador de Corrente	23
Figura 12 - Amplificador Vcontrole	23
Figura 13 - Circuito de comutação Man/aut.....	24
Figura 14 - Sensor de Temperatura	25
Figura 15 - Circuito de acionamento do Cooler	26
Figura 16 - Circuito Inibidor	27
Figura 17 - Fonte de alimentação linear 12V, 5V /1A.....	27
Figura 18 - Diagrama em blocos	28
Figura 19 - Visualização do Lay-out	29
Figura 20 - Visualização Placa base	30
Figura 21 - Gravador PICKIT 3.....	31
Figura 22 - Rotina Principal	32
Figura 23 - Rotina Seleção Modo de Operação	33
Figura 24 - Rotina Ajuste de SetPoint	34
Figura 25 - Placa Módulo de potência.....	35
Figura 26 - Vista superior do Módulo de Potência.....	36
Figura 27 - Vista inferior do Módulo de Potência.....	36
Figura 28 - Módulo de potência finalizado.....	37
Figura 29 - Vista inferior da Placa Base	37

Figura 30 - Vista Superior da Placa Base Montada.....	38
Figura 31 - Vista inferior da Placa de Controle.....	39
Figura 32 - Placa de Controle montada.....	40
Figura 33 - Atividades Hardware	43
Figura 34 - Atividades Mecânica	43
Figura 35 - Atividades Firmware.....	44
Figura 36 - PERT Carga Eletrônica.....	45
Figura 37 - GANT CARGA ELETRÔNICA	46
Figura 38 - EAP Carga Eletrônica	48

Sumário

1	INTRODUÇÃO.....	13
1.1	OBJETIVO	13
1.2	ESTADO DA ARTE	14
2	DESENVOLVIMENTO	15
2.1	PROJETO DE CONSTRUÇÃO	15
2.2	MÓDULO DE POTÊNCIA	15
2.3	CIRCUITO DE CONTROLE E GERENCIAMENTO	19
2.3.1	<i>Microcontrolador.....</i>	<i>20</i>
2.3.2	<i>Teclado</i>	<i>21</i>
2.3.3	<i>Display</i>	<i>22</i>
2.3.4	<i>Amplificador de corrente</i>	<i>22</i>
2.3.5	<i>Amplificador Vcontrole</i>	<i>23</i>
2.3.6	<i>Circuito de Comutação Man/Aut.....</i>	<i>24</i>
2.3.7	<i>Sensor de Temperatura</i>	<i>25</i>
2.3.8	<i>Circuito de Acionamento do cooler.....</i>	<i>25</i>
2.3.9	<i>Circuito Inibidor</i>	<i>26</i>
2.3.10	<i>Fonte de alimentação.....</i>	<i>27</i>
2.3.11	<i>Diagrama em Blocos</i>	<i>28</i>
2.4	PLACA BASE	30
2.5	FIRMWARE	31
2.5.1	<i>Fluxograma Rotina principal.....</i>	<i>32</i>
2.5.2	<i>Fluxograma Seleção de Modo de Operação.....</i>	<i>33</i>
2.5.3	<i>Fluxograma Ajuste de SetPoint.....</i>	<i>34</i>
2.6	CONFECÇÃO DAS PLACAS	35
2.6.1	<i>Placa Módulo de Potência.....</i>	<i>35</i>
2.6.2	<i>Placa Base</i>	<i>37</i>
2.6.3	<i>Placa de Controle.....</i>	<i>39</i>
3	CONCLUSÃO.....	41
	REFERÊNCIAS.....	42
	APÊNDICE A	43

A.1 PERT	43
A.2 GANT	46
A.3 EAP (ESTRUTURA ANALÍTICA DO PROJETO)	47
APÊNDICE B	51
B.2 ROTINA GRAVAÇÃO	51
B.3 ROTINA LEITURA	57
B.4 ROTINA LCD	57
B.5 ROTINA MODO DE OPERAÇÃO	59
B.6 ROTINA SELEÇÃO DE MODO	65
B.7 ROTINA SELEÇÃO VALOR	67

1 INTRODUÇÃO

Este trabalho trata da construção de uma Carga Eletrônica Ajustável para corrente contínua, que terá o ajuste de 0 a 360W, na qual fornece maior versatilidade e produtividade ao testar cargas (Fontes de Alimentação DC, Baterias, Painéis Solares). Assim a pesquisa norteou-se, partindo da situação na qual, a indústria testa as cargas por meio de banco de resistores, com intuito de conseguir puxar a máxima corrente fornecida sendo necessária a associação de resistores, a maioria das vezes de alta potência e de grandes dimensões físicas.

A pesquisa tem como justificativa, apresentar outro método além do banco de resistores, para testar as cargas. De simples modo, é baseado na característica do transistor de se comportar como uma resistência ativa quando polarizados adequadamente. Isto possibilita variar a resistência interna deste componente, variando de maneira controlada a carga aplicada ao equipamento em teste.

1.1 OBJETIVO

O projeto em questão diz respeito a implementação de um equipamento que simule um resistor ativo onde através de um controle acessível ao usuário forneça uma carga ajustável que varie de 0 a 360W para aplicação de testes e aferição das especificações elétricas de equipamentos geradores, conversores ou armazenadores de energia.

1.2 ESTADO DA ARTE

Cargas eletrônicas ou também conhecidas como cargas dinâmicas são dispositivos usados para atender as necessidades do mercado, onde é necessário o teste de equipamentos ou produtos fornecedores de tensão contínua (CC). Estes que precisam de cargas sendo elas fixas, dinâmicas ou pré-definidas.

As cargas Eletrônicas, são usadas para testes dos principais tipos de equipamento:

- a. Teste de fonte de alimentação DC
- b. Baterias
- c. Carregadores de bateria
- d. Conversores DC-DC,
- e. Painéis solares

Equipamentos como estes, antes dos desenvolvimentos de ferramentas mais modernas e com uma variedade maior de recursos, eram usados outros métodos para testes individuais para cada necessidade requerida, tornando o processo mais demorado e específico de cada equipamento.

Com o surgimento de novas ferramentas, métodos mais modernos e eficazes foram sendo utilizados, dentre eles a “Carga Eletrônica” surgiu para substituir métodos convencionais e tornando-os mais práticos e eficazes, sendo capaz de atender a maiores necessidades e operações com um único aparelho.

Existem poucos equipamentos voltados para esses tipos de testes, do qual forneçam todos os recursos a fim de atender de melhor forma a necessidade de cada equipamento, e suas peculiaridades.

Pensando nisso empresas como “**BK Precision**” e “**Chroma**”, que atualmente são as empresas consolidadas no mercado, vem fornecendo os mais diversos equipamentos e modelos, atendendo o mercado de maneira eficaz.

2 DESENVOLVIMENTO

2.1 PROJETO DE CONSTRUÇÃO

O projeto carga eletrônica é um Equipamento destinado a testar equipamentos geradores ou fornecedores de energia CC. Tem a principal característica de consumir a energia gerada/fornecida por estes equipamentos num processo de drenagem da corrente dos mesmos. Para isto conta com 06 células de potência, cada uma consumindo no máximo 60W perfazendo um total de 360W.

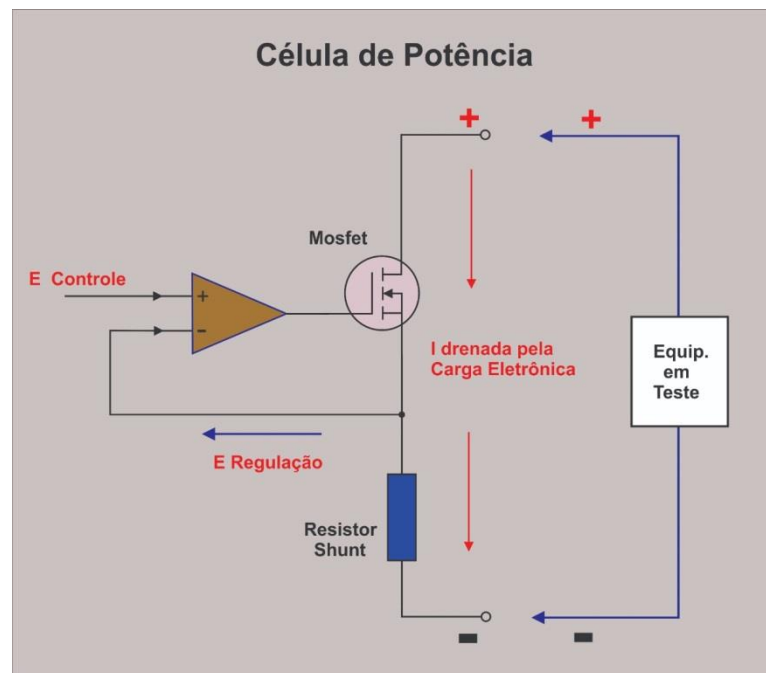
2.2 MÓDULO DE POTÊNCIA

O projeto contém 6 células de potência funcionando de forma paralela.

Conforme pode ser visto na figura 1, cada célula de potência é composta por, um mosfet IRFP 264, um resistor shunt e um amplificador operacional, funcionando na forma de comparador.

Ao aplicarmos uma tensão de controle na entrada não inversora do amplificador, o mesmo irá injetar uma tensão no gate do mosfet fazendo este conduzir. Desta maneira, uma corrente é drenada do equipamento que está sob teste. Assim, a corrente ao passar pelo resistor shunt, gera uma tensão em seus terminais, que é injetada na entrada inversora do comparador, fazendo assim o controle da corrente que está sendo drenada.

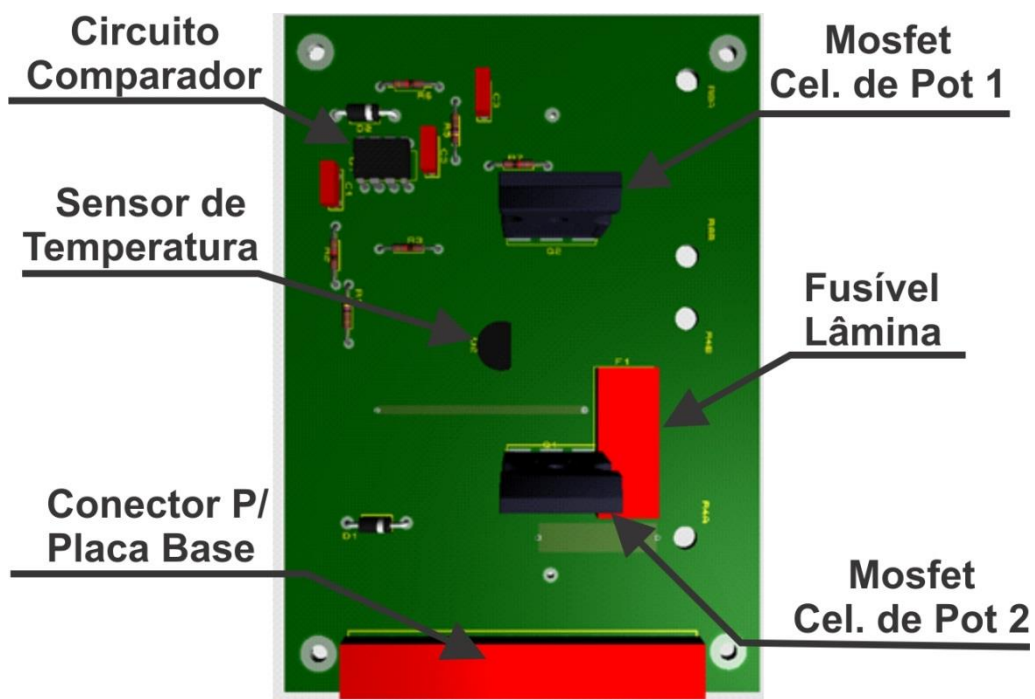
Figura 1 - Célula de Potência



Fonte: Próprio Autor

Na figura 2, pode-se observar a ilustração de um dos módulos de potência contendo 02 células de potência de 60W cada (O projeto contempla 03 módulos de potência com 02 células cada).

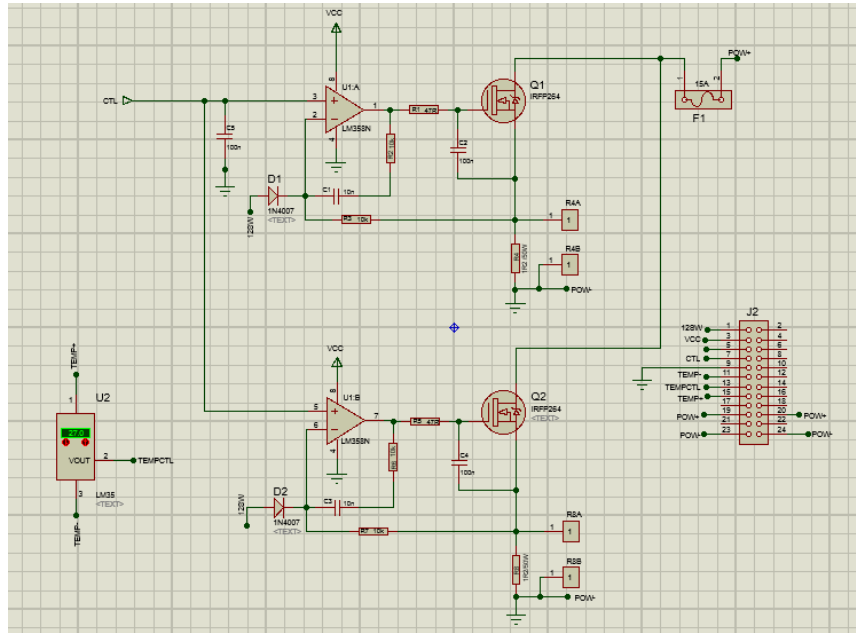
Figura 2 - Módulo de Potência



Fonte: Próprio autor (Proteus)

Na figura 3, pode-se observar o Diagrama Esquemático do Módulo de Potência.

Figura 3 - Diagrama Esquemático Módulo de Potência



Fonte: Próprio autor (Proteus)

Na figura 4, pode-se observar o elemento consumidor de potência, IRFP264, que é um MOSFET de potência que suporta tensões de até 250 V e corrente de no máximo 38A. É por este semicondutor que será drenada toda a corrente do equipamento sob teste.

Figura 4 - Mosfet IRFP264



VDS – 250V
VGS – 20V
RDS – 0,075Ω
ID – 38A

Fonte: <https://pt.aliexpress.com/item/33037407382.html> 28/06/20

Para o controle de condução do Mosfet é utilizado o amplificador operacional LM358, conforme pode ser visto na figura 5. Este por sua vez injetará através de sua saída a tensão necessária para proporcionar a condução do elemento de potência.

Em uma de suas entradas será aplicado o feedback da corrente drenada pelo elemento de potência enquanto na outra entrada será aplicada a tensão de controle necessária para condução do Mosfet.

Figura 5 - Amplificador Operacional LM 358

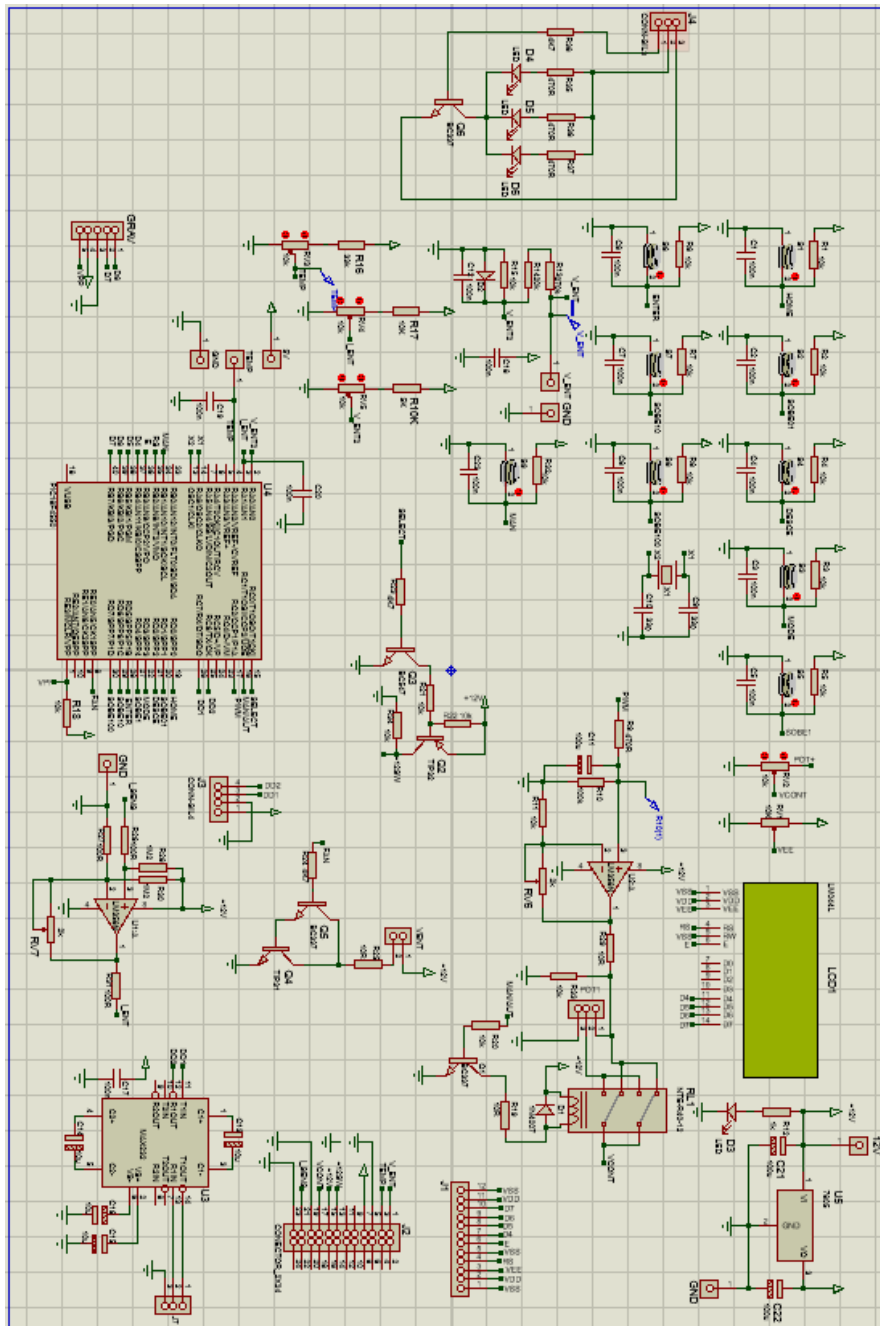


Fonte: <https://envirementalb.com/lm358-ic-pinout-description/29/06/20>

2.3 CIRCUITO DE CONTROLE E GERENCIAMENTO

Na figura 6, pode-se observar o diagrama esquemático do Circuito de Controle e gerenciamento. Este é o circuito responsável por gerar o sinal PWM que controlará os módulos de potência, além de monitorar e tratar os sinais de tensão, corrente e temperatura provenientes de sensores localizados no módulo de potência.

Figura 6 - Circuito de Controle e gerenciamento



Fonte: Próprio autor (Proteus)

2.3.1 Microcontrolador

O circuito de controle e gerenciamento é composto por um microcontrolador PIC 18F4550 que recebe os sinais em seus conversores A/D e os converte, realizando cálculos para tomadas de decisão e controle do PWM, sendo este o componente principal para gestão de todo o processo de controle e monitoramento da Carga Eletrônica.

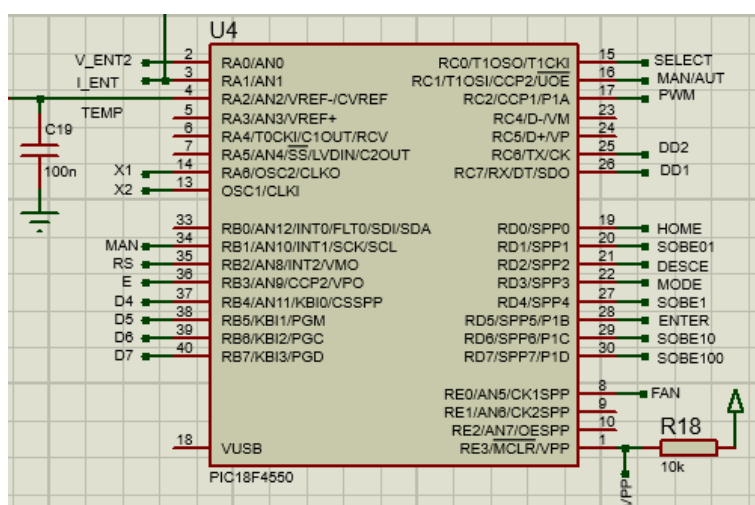
Também recebe os dados para set-point digitados através de um teclado, além de escrever em um display LCD informações inerentes à operação do equipamento para visualização pelo usuário. Na figura 7 pode-se observar a ilustração do Microcontrolador em questão e na figura 8 o Diagrama elétrico com a identificação dos pinos utilizados:

Figura 7 - Microcontrolador PIC18F4550



Fonte: <https://shopee.com.br/Microcontroladores-USB-PIC18F4550-I-P-PIC18F4550-IC-DIP40-IC-PIC-MCU-FLASH-16KX16-i.191541155.4604864938> (29/06/2020)

Figura 7 - Diagrama Elétrico PIC18F4550



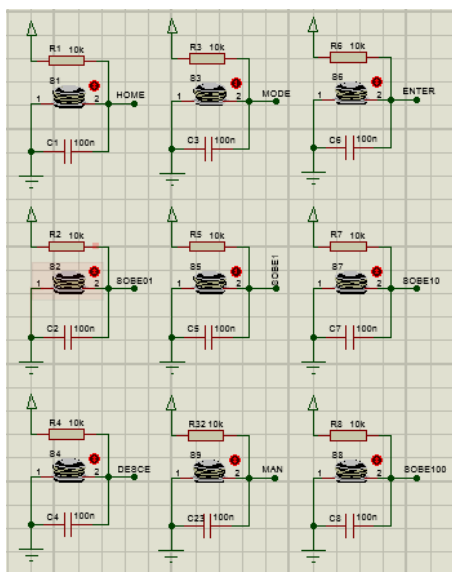
Fonte: Autor (Proteus)

2.3.2 Teclado

O teclado é composto por nove chaves do tipo tátil através das quais o operador poderá selecionar os modos de operação e configurar valores de setpoint. Pode-se também gravar ou apagar valores como tempo de operação e energia consumida da memória EEPROM.

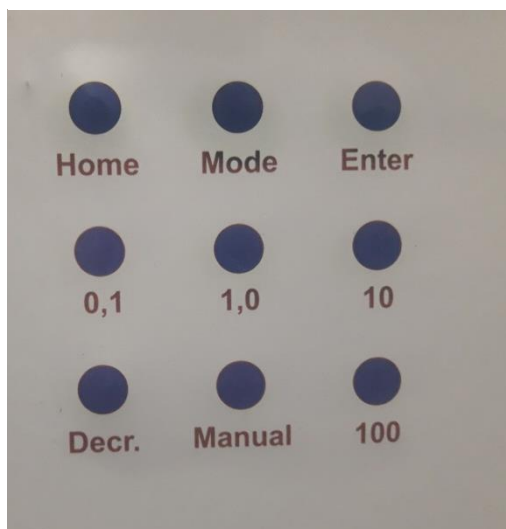
Na figura 9 pode-se observar o diagrama esquemático das chaves e suas funções e na figura 10 a disposição final no Painel da Carga Eletrônica.

Figura 8 - Diagrama Esquemático do Teclado



Fonte: Próprio autor (Proteus)

Figura 9 - Teclado no Painel Frontal



Fonte: Próprio autor

2.3.3 Display

O projeto Carga Eletrônica CC também contempla um display LCD para informar ao usuário o modo de operação selecionado, o valor de setpoint escolhido, a tensão aplicada na entrada da Carga Eletrônica, a corrente que está sendo drenada, a energia que está sendo consumida, o valor do duty cycle do PWM para tensão de controle, o tempo de operação da Carga Eletrônica e a temperatura dos módulos de potência.

O Display LCD utilizado é um display de 4 linhas por 20 colunas modelo LCM 2004D de background azul, conforme pode-se observar na figura 11.

Figura 10 - Display



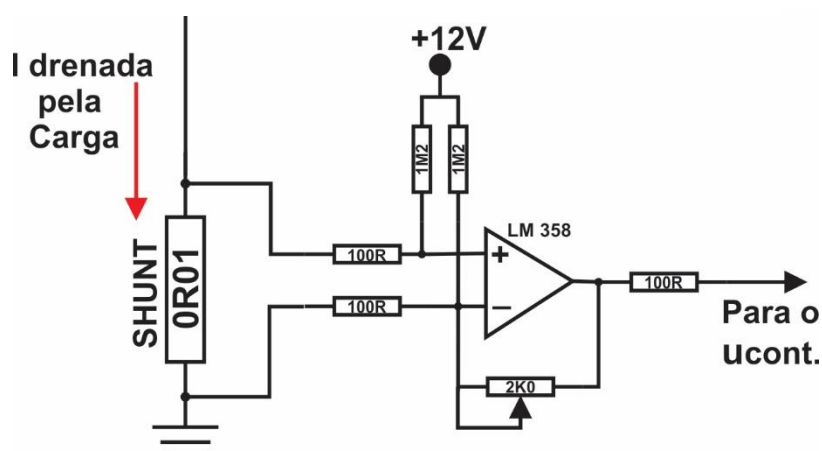
Fonte: Próprio autor

2.3.4 Amplificador de corrente

O amplificador de corrente é composto basicamente pelo CI LM358 cuja função é proporcionar um ganho de tensão ao sinal proveniente do shunt de leitura de corrente geral.

Conforme pode-se observar na figura 12, a corrente drenada pelos seis módulos passa pelo resistor shunt e gera uma tensão proporcional em seus terminais. Esta tensão é aplicada ao amplificador de corrente, que a eleva ao nível necessário para ser aplicado ao conversor A/D do microcontrolador. O ganho deste amplificador é da ordem de dez vezes, ou seja, para cada 1 ampere de corrente drenada tem-se na saída do operacional, aproximadamente 100mV.

Figura 11 - Amplificador de Corrente

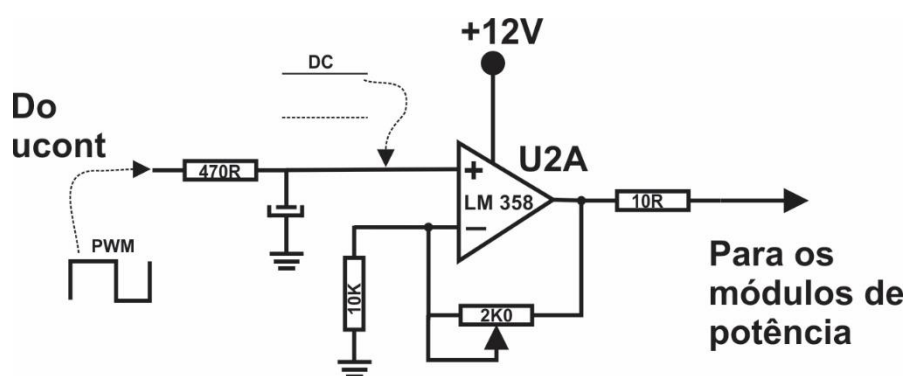


Fonte: Próprio autor

2.3.5 Amplificador Vcontrole

O sinal PWM entregue pelo microcontrolador, passa por um circuito integrador, convertendo-se em uma tensão de controle DC. Após, esta tensão é entregue ao amplificador operacional (LM 358), onde sofrerá um pequeno ganho de tensão e corrente para então ser entregue às células de potência. Na figura 12 pode-se observar o diagrama esquemático deste bloco.

Figura 12 - Amplificador Vcontrole



Fonte: Próprio autor

2.3.6 Circuito de Comutação Man/Aut.

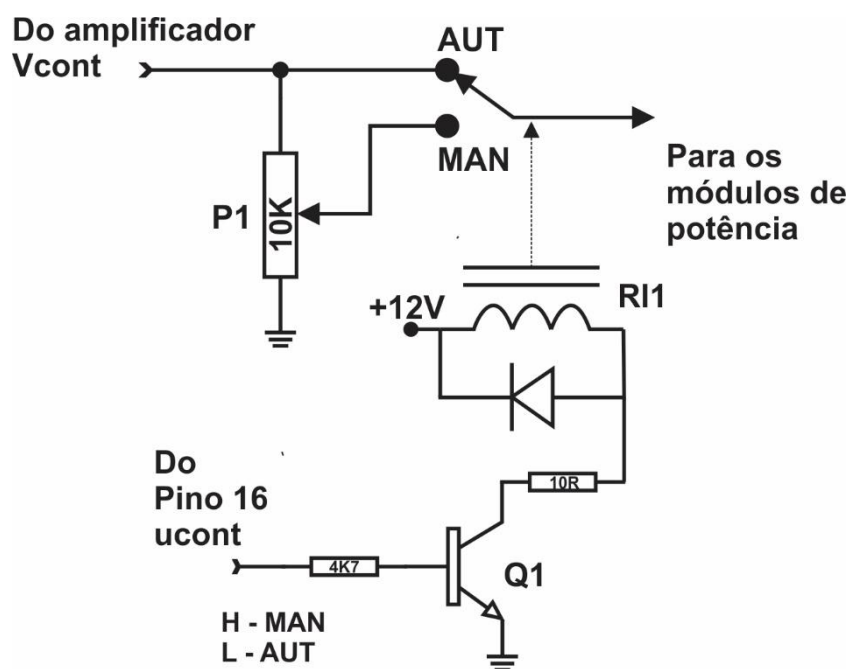
A tensão de controle que será aplicada aos módulos de Potência será aplicada aos contatos de um relé. Este relé é que irá fazer a comutação para o modo Automático ou Manual.

O modo Automático será comutado quando os modos de operação de Tensão Constante ou Potência constante ou Resistência Constante ou Corrente Constante forem selecionados. Nesta condição o relé estará desenergizado e a V de controle será aplicada diretamente nas células de Potência, polarizando os Gates dos Mosfet's, fazendo a corrente ser drenada.

Caso seja selecionado o modo Manual, o pino 16 do microcontrolador vai para nível alto, fazendo o transistor Q1 conduzir, energizando O relé RL1 que comuta para o outro contato. Com isto, a V de controle é desviada par o Potenciômetro P1 e desta forma a tensão de controle pode ser graduada pelo usuário antes de ir para as Células de Potência e assim permitir controlar de forma manual a corrente a ser drenada.

Na figura 13 pode-se observar o diagrama do Circuito de Comutação

Figura 13 - Circuito de comutação Man/aut.



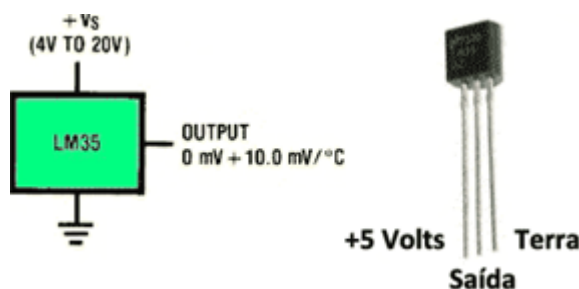
Fonte: Próprio autor

2.3.7 Sensor de Temperatura

Nos blocos de potência ocorrerá os aquecimentos dos elementos dissipadores acoplados aos MOSFET's e, portanto, a temperatura dos mesmos estará sendo monitorada por um sensor de temperatura.

O sensor de temperatura utilizado é o LM 35 que irá captar a temperatura nos dissipadores dos blocos de potência e a converterá em tensão, enviando para o pino 4 do microcontrolador o qual, ao detectar determinado nível acionará o cooler, e/ou caso ocorra um superaquecimento ultrapassando os níveis de segurança para o equipamento, inibirá o PWM e abortará o ciclo de operação sinalizando para o operador a condição presente através do display. Na figura 14 pode-se observar a foto ilustrativa do sensor de temperatura utilizado:

Figura 14 - Sensor de Temperatura



Fonte:
(29/06/2020)

<http://blog.novaeletronica.com.br/lm35-o-sensor-de-temperatura-mais-popular/>

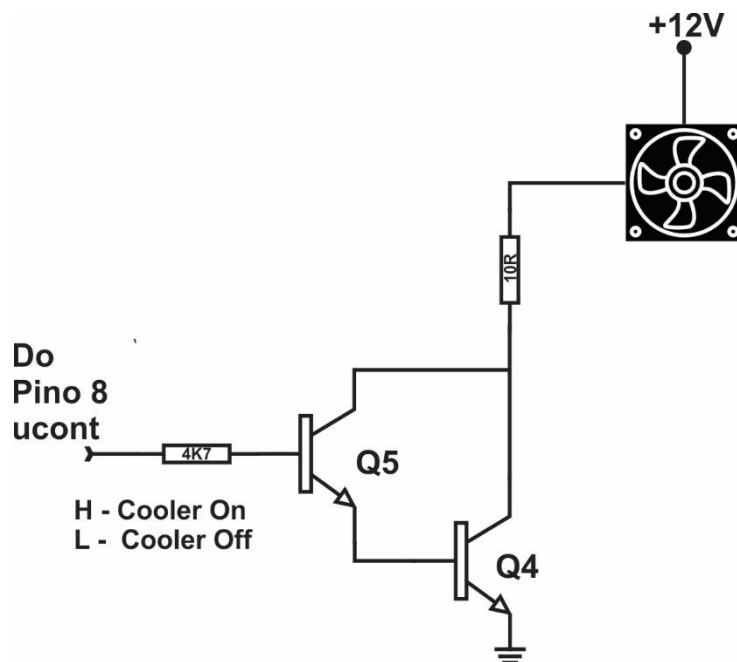
2.3.8 Circuito de Acionamento do cooler

Para se evitar o superaquecimento dos blocos de potência o projeto contempla a utilização de um cooler para promover o resfriamento dos dissipadores, os quais estão devidamente posicionados no interior do gabinete, próximo aos blocos de potência.

Ao se inicializar a carga eletrônica o cooler se encontra desligado. Caso a potência de trabalho da carga eletrônica exceder os 30W ou a temperatura atingir 40°C o pino 8 do microcontrolador vai para nível alto, Com isto a base do transistor Q5 é polarizada fazendo-o conduzir, o que provoca a saturação de Q4 e por sua vez o acionamento do cooler. Uma vez acionado, o cooler só será desligado caso a temperatura resfrie abaixo de 30°C e a potência de trabalho seja inferior a 28W.

Na figura 15 pode-se observar o circuito responsável pelo acionamento do cooler.

Figura 15 - Circuito de acionamento do Cooler



Fonte: Próprio autor

2.3.9 Circuito Inibidor

O projeto Carga eletrônica funciona com seis células de potência ligadas em paralelo, cada uma drenando no máximo 5A, totalizando 30A quando as seis estão em funcionamento.

A tensão de controle polariza as seis células simultaneamente, desta forma, ao selecionar o valor da corrente a ser drenada o software divide o valor setado por 6 e ajusta o PWM de tal forma que a tensão de controle gerada para excitar as células de potência represente 1/6 do total e como todas estão em paralelo o resultado final da corrente total drenada será o valor desejado.

Isto funcionou muito bem quando a corrente desejada é acima de 4A, porém quando é muito baixa, perde-se resolução por conta da divisão por seis.

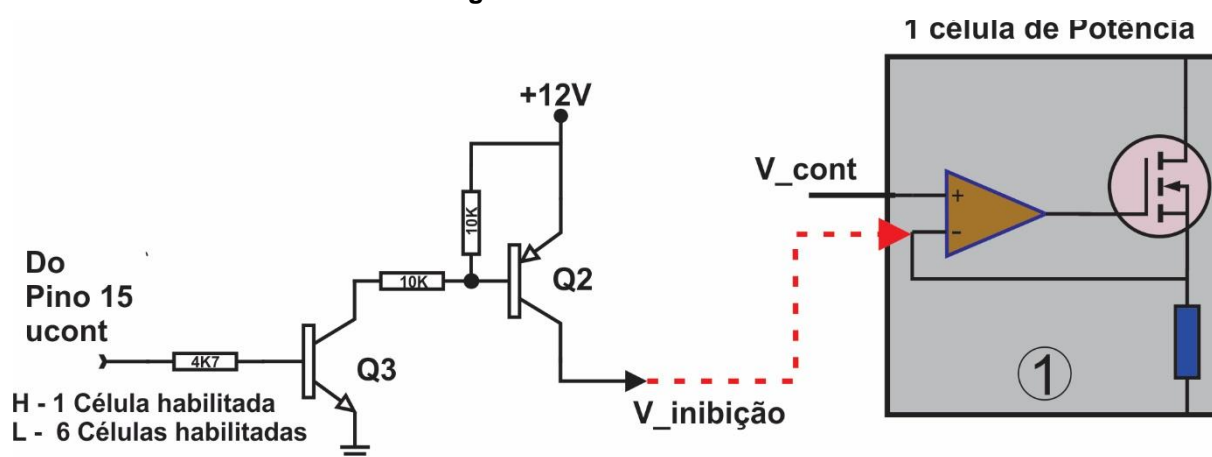
Para contornar este problema foi acrescentada uma rotina no software que quando o valor setado é acima de 4A, o processo corre normal, a divisão por seis é realizada e as seis células estão ativas. Caso o valor setado seja inferior a 4A, não ocorrerá a divisão por seis e apenas uma célula de potência permanece ativa proporcionando assim uma melhor resolução do PWM e, por conseguinte da tensão de controle que irá excitar esta célula.

A nível de hardware, foi implementado um circuito, que ao ser ativado pelo pino 15 do microcontrolador faz com que Q3 sature, provocando a saturação de Q2. Com Q2 saturado, a tensão de 12V é aplicada às entradas inversoras de cinco das seis células de potência inibindo o seu funcionamento, ficando apenas uma célula de

potência em atividade. Caso a corrente setada seja superior á 4A, o pino 15 do microcontrolador permanece em nível baixo, o circuito de inibição não é acionado e todas as seis células estão ativas.

Na figura 16 pode-se observar o circuito aqui descrito.

Figura 16 - Circuito Inibidor

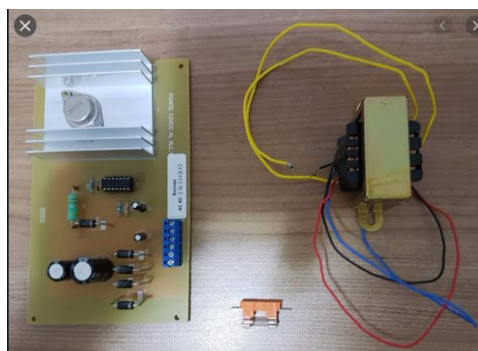


Fonte: Próprio autor

2.3.10 Fonte de alimentação

Para alimentar todo o circuito do projeto como microcontrolador, periféricos e circuitos auxiliares é utilizada uma Fonte de alimentação linear. Esta fonte é composta por um transformador abaixador de tensão, diodos retificadores, filtros capacitivos, e circuitos reguladores de tensão. Fornece tensões reguladas de 5V e 12V com corrente máxima de 2A. Na figura 17 pode-se observar a foto ilustrativa da fonte utilizada.

Figura 17 - Fonte de alimentação linear 12V, 5V /1A

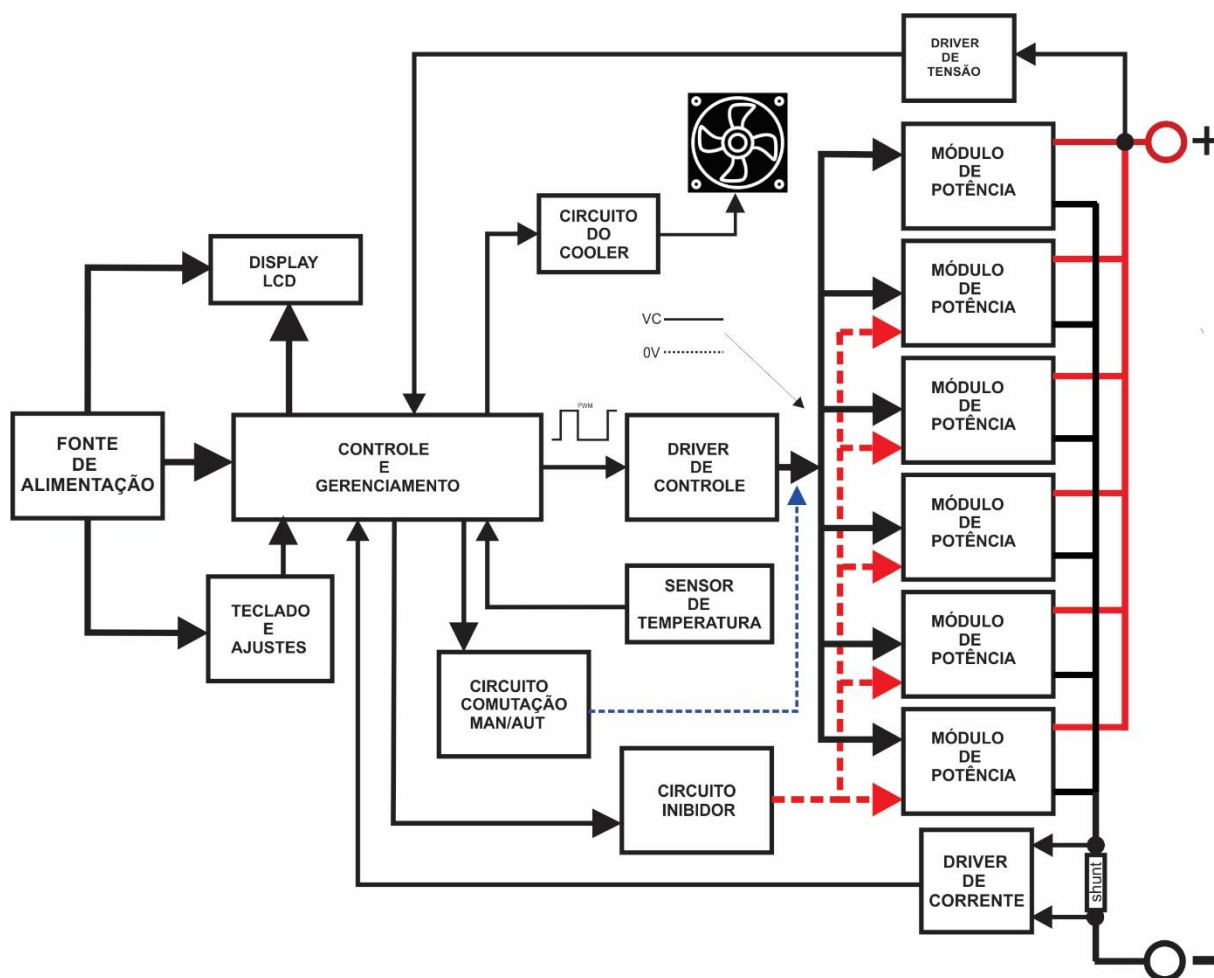


Fonte: <https://produto.mercadolivre.com.br/MLB-861138157-fonte-de-alimentaco-linear-estabilizada-12-v-3-a- JM?quantity=1> (29/07/2020)

2.3.11 Diagrama em Blocos

Na figura 18, pode-se observar o diagrama em blocos mostrando uma visão geral da carga eletrônica e de todos os circuitos que a compõe.

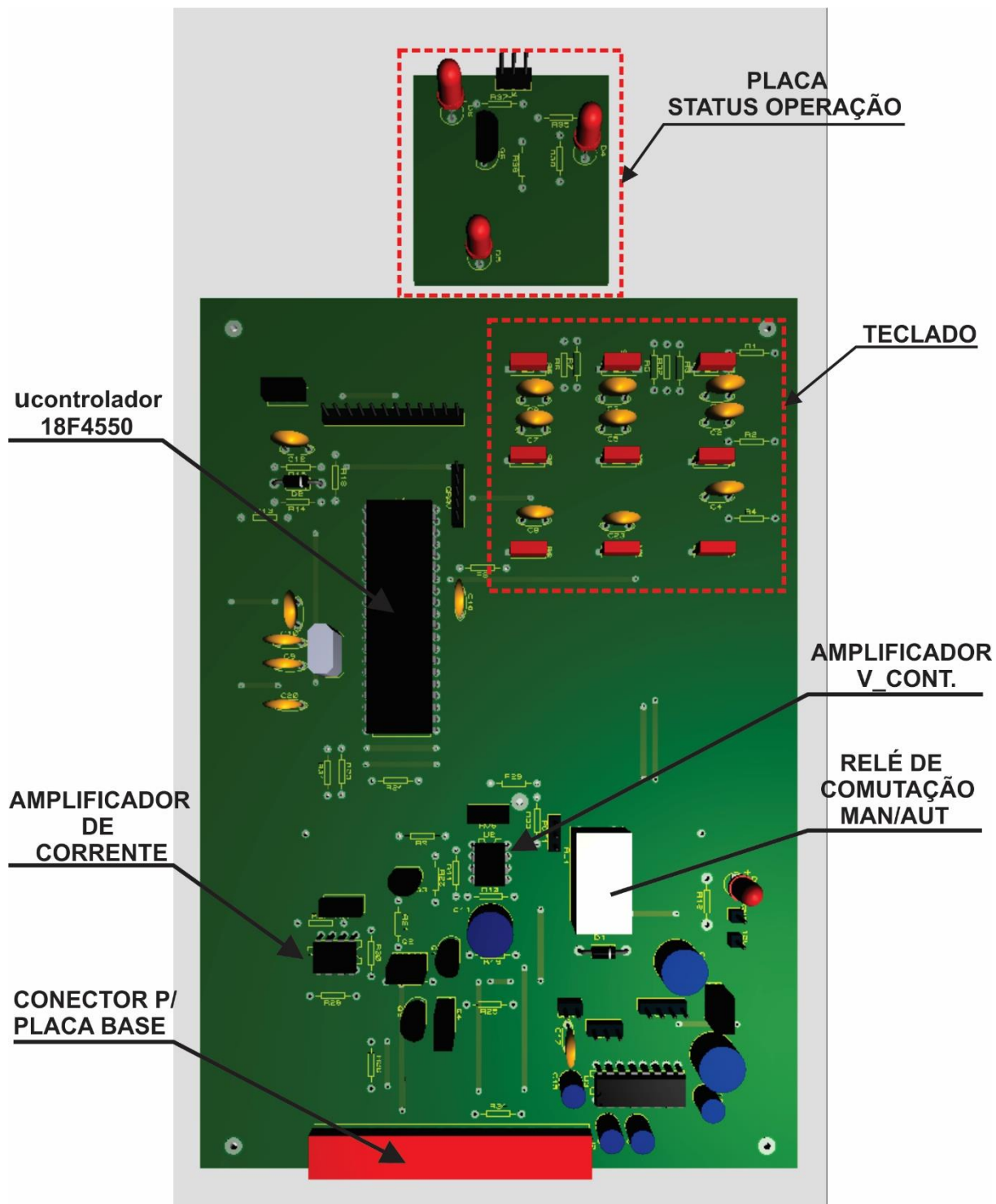
Figura 18 - Diagrama em blocos



Fonte: Próprio autor

Na figura 19, pode-se observar a visualização em 3D do software proteus utilizado para desenvolver o Lay-out da placa de controle. É possível visualizar todo o aspecto final e disposição dos componentes na placa.

Figura 19 - Visualização do Lay-out

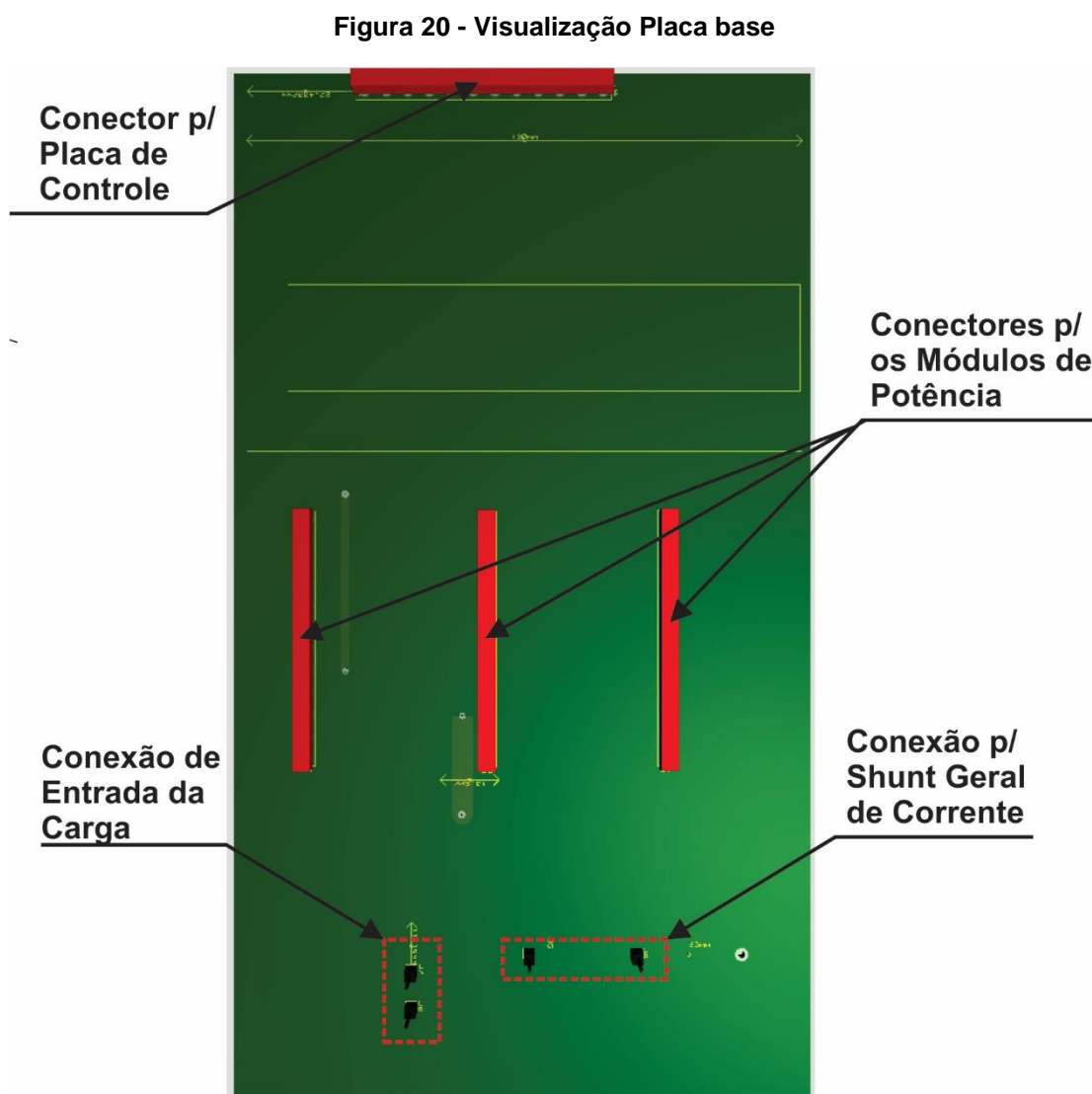


Fonte: Próprio autor (Proteus)

2.4 PLACA BASE

O projeto contempla três Módulos de Potência e uma placa de controle. As ligações elétricas entre estas diferentes placas necessitam de uma razoável quantidade de fiação, o que torna a montagem bastante complexa e o aspecto final não fica agradável. Visando contornar este problema e tornar a montagem mais limpa e o aspecto final mais agradável, foi pensado no projeto a introdução de uma terceira placa, através da qual seja feita a interligação entre os módulos de potência e a placa de controle.

Na figura 20, pode-se observar o lay-out desta placa e as disposições e distribuições dos conectores em sua superfície.



Fonte: Próprio autor (Proteus)

2.5 FIRMWARE

O código desenvolvida para fazer a gestão e controle do projeto Carga Eletrônica foi feito em linguagem C e o compilador utilizado foi o CCS C Compiler.

O processo de criação do código foi sendo desenvolvido em partes e sendo testado também parcialmente em circuito real montado em protoboard. Sendo assim, primeiramente desenvolveu-se o código para gerar o PWM, após o código para fazer a leitura de tensão em seguida, foi feito o código para apresentar a informação em um display e assim por diante. Após, numa segunda fase do processo foi feita a integração destes códigos formando assim um código único. Na terceira fase do processo o código foi ganhando robustez e sendo acrescentado rotinas para alarmes, proteções e armazenamento e leitura de dados.

O resultado final foi bastante satisfatório, atendendo as necessidades do trabalho aqui apresentado e atingido as expectativas quanto á eficácia do projeto como um todo.

A seguir, será descrito as rotinas do código em fluxograma e o código em C na íntegra pode ser acessado nos anexos.

Para gravação do código no microcontrolador utilizou-se o gravador PICKIT 3 conforme pode ser observado na figura 21.

Figura 21 - Gravador PICKIT 3

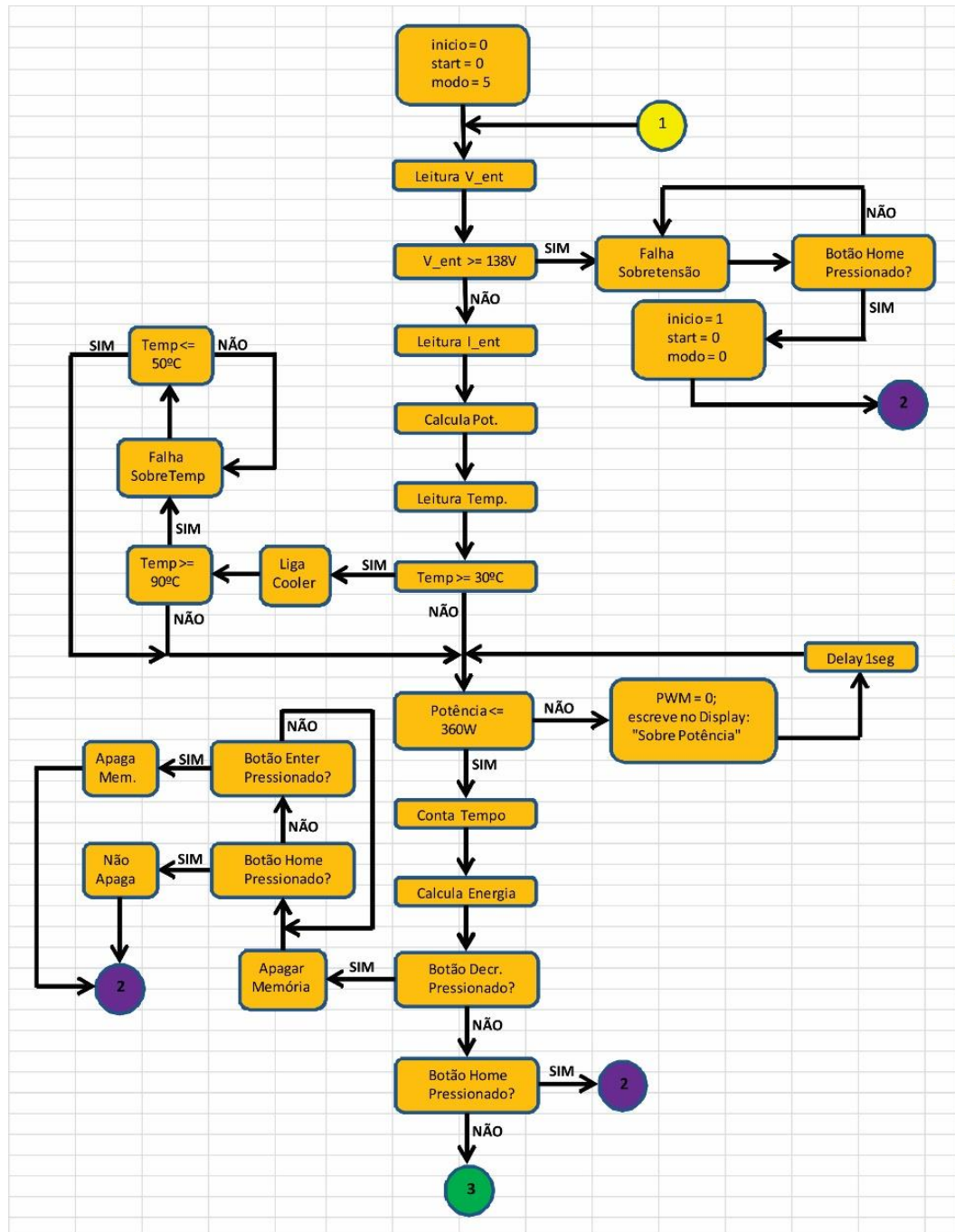


Fonte: Próprio autor

2.5.1 Fluxograma Rotina principal

Na figura 22, pode-se observar a rotina principal do código desenvolvido. Aqui, enquanto o código estiver rodando, ele sempre passará por esta rotina a cada looping realizado.

Figura 22 - Rotina Principal

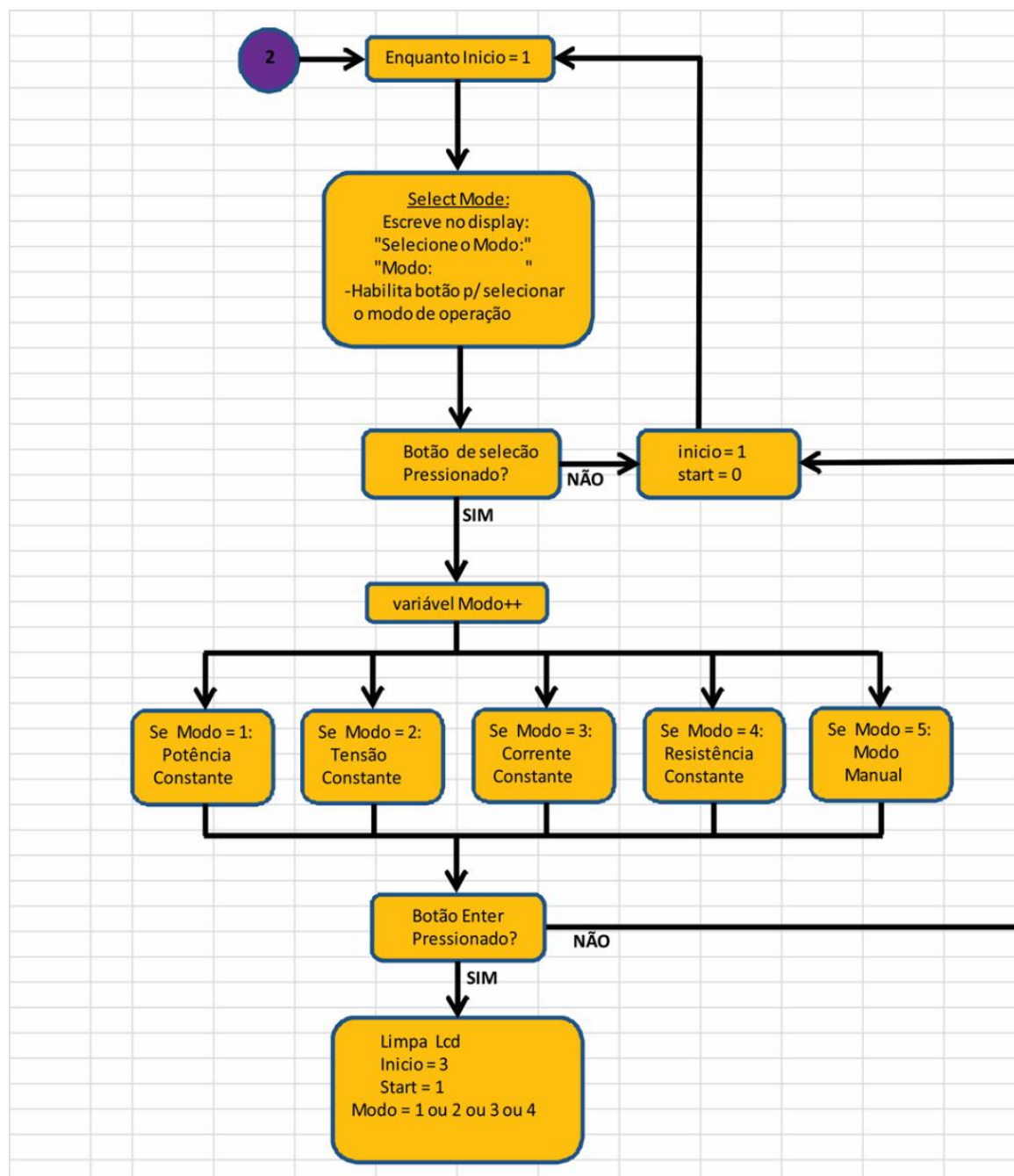


Fonte: Próprio autor

2.5.2 Fluxograma Seleção de Modo de Operação

Na figura 23 pode-se observar a rotina para seleção do modo de operação da carga Eletrônica. Nesta rotina, o usuário pode selecionar através do teclado no painel o modo de operação que se deseja trabalhar

Figura 23 - Rotina Seleção Modo de Operação

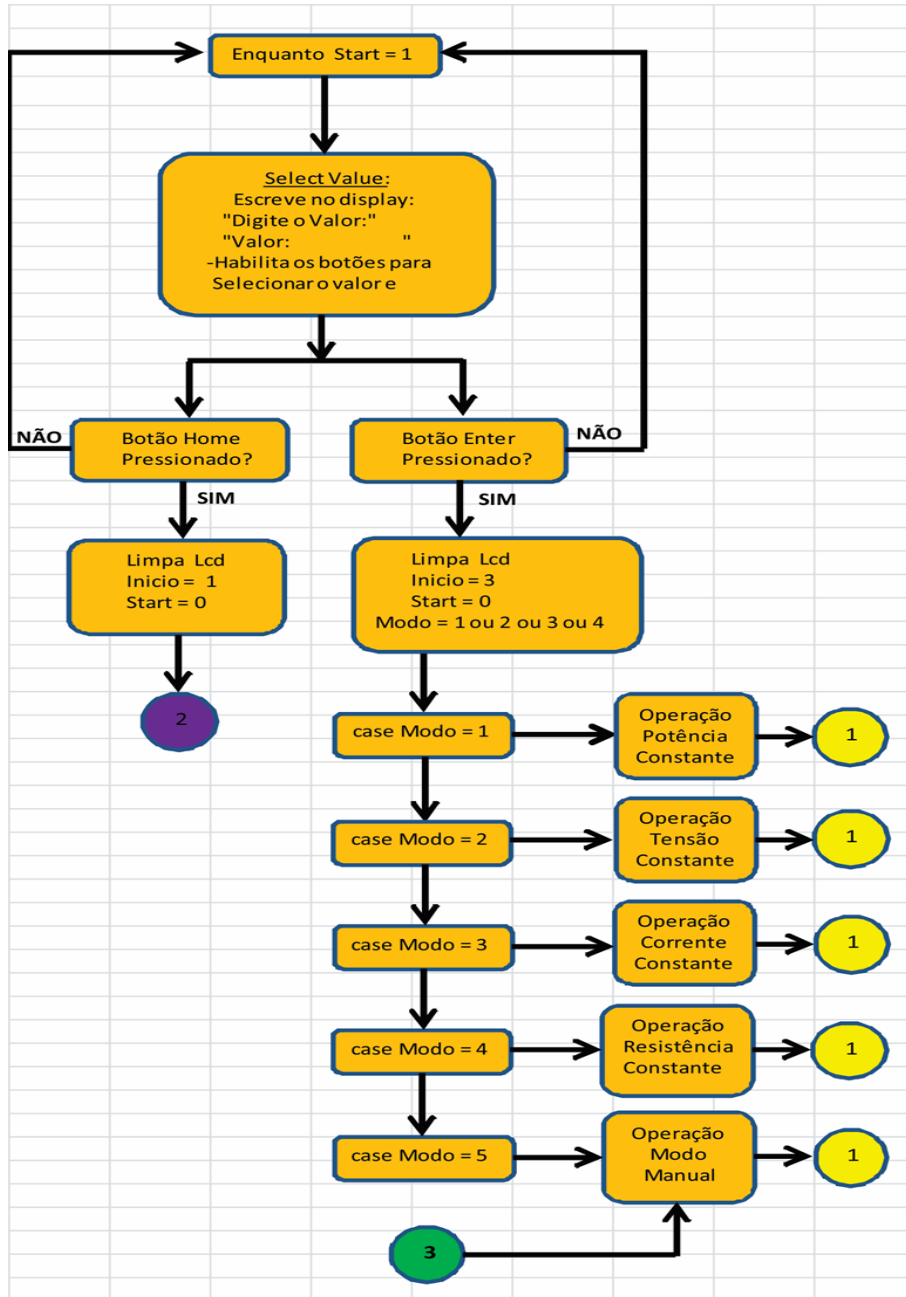


Fonte: Próprio autor

2.5.3 Fluxograma Ajuste de SetPoint

Na figura 24, pode-se observar a rotina para seleção dos valores de Setpoint que se deseja trabalhar para um determinado modo de operação escolhido anteriormente. Uma vez selecionado o modo de operação desejado, o processo cai nesta rotina, onde permite ao usuário configurar através do teclado no painel os valores que se deseja trabalhar.

Figura 24 - Rotina Ajuste de SetPoint



Fonte: Próprio autor

2.6 CONFECÇÃO DAS PLACAS

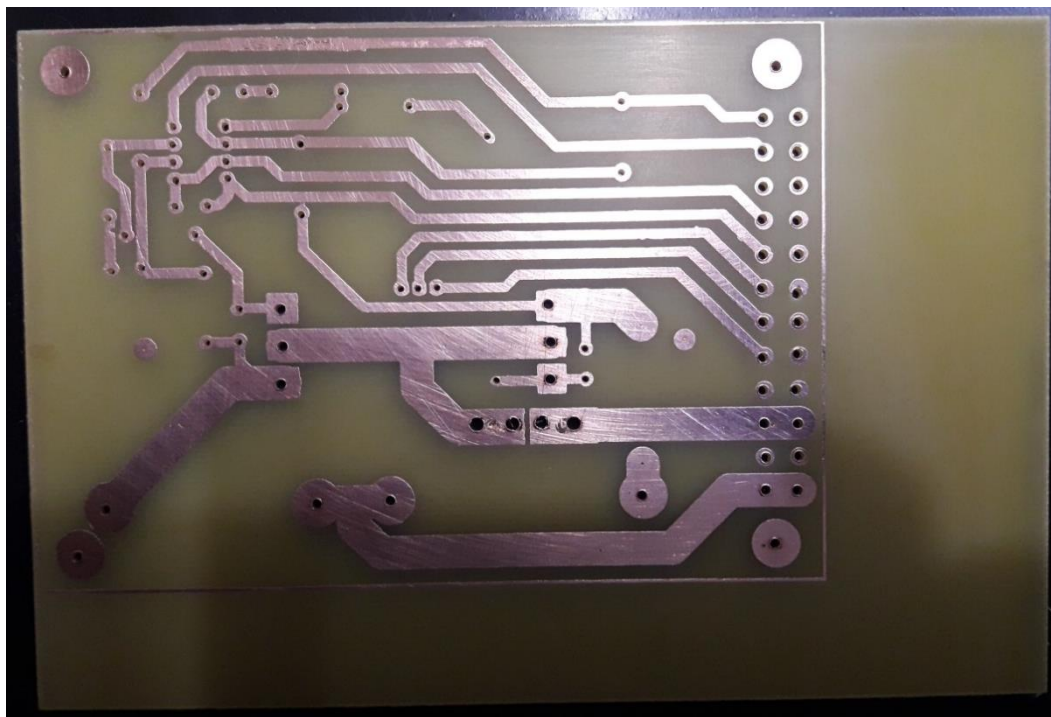
O projeto Carga eletrônica é composto de cinco placas sendo três para os módulos de potência, uma placa base e uma placa de Controle.

Todas as placas foram confeccionadas no processo de transferência térmica, onde o Lay-out é impresso a laser sobre um papel fotográfico e posteriormente é transferido por aquecimento para a placa cobreada, sendo então mergulhada em uma solução de percloreto de ferro para sofrer o processo de corrosão.

2.6.1 Placa Módulo de Potência

Na figura 26 pode-se observar a placa do módulo de potência após a impressão e corrosão.

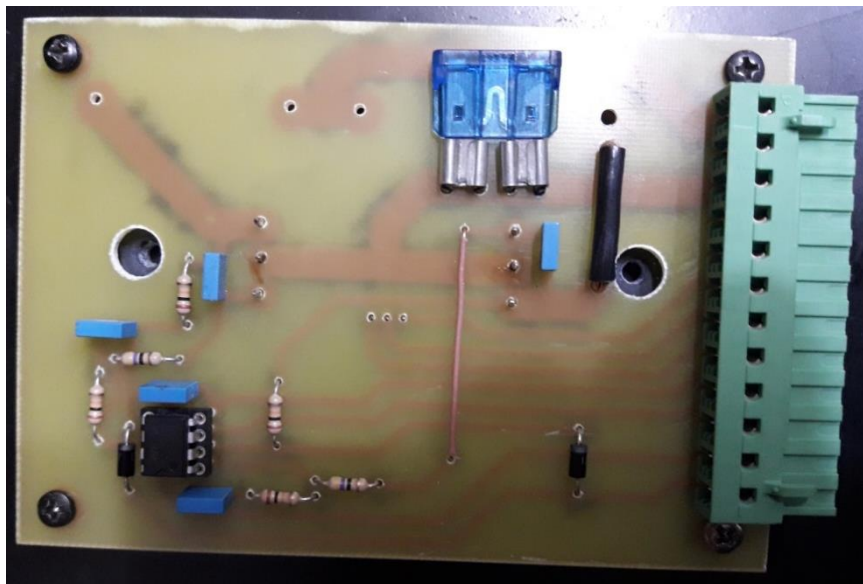
Figura 25 - Placa Módulo de potência



Fonte: Próprio autor

Na figura 26 pode-se observar a placa do módulo de potência já com os componentes soldados na mesma.

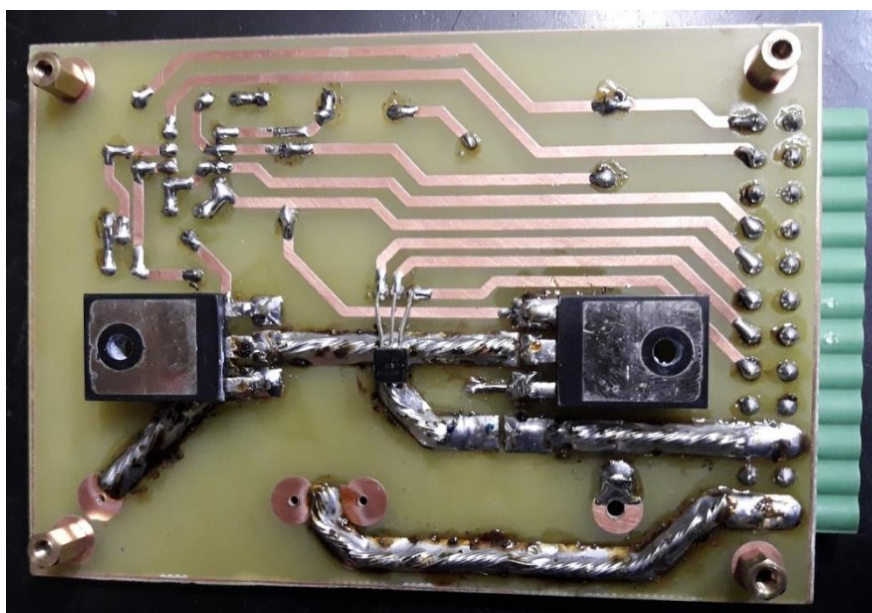
Figura 26 - Vista superior do Módulo de Potência



Fonte: Próprio autor

Na figura 27 pode-se observar a vista inferior da placa do módulo de potência

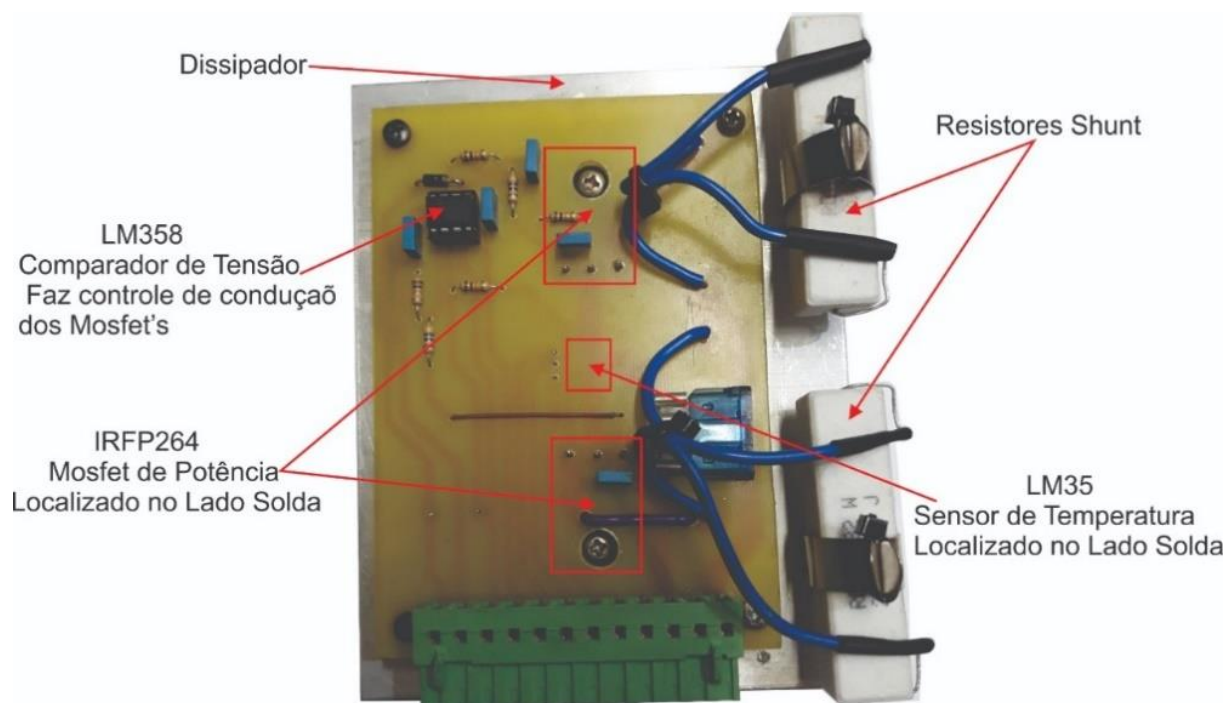
Figura 27 - Vista inferior do Módulo de Potência



Fonte: Próprio autor

Na figura 28 pode-se observar a placa do Módulo de Potência já montado com todos os componentes inclusive com os resistores shunt e o dissipador.

Figura 28 - Módulo de potência finalizado

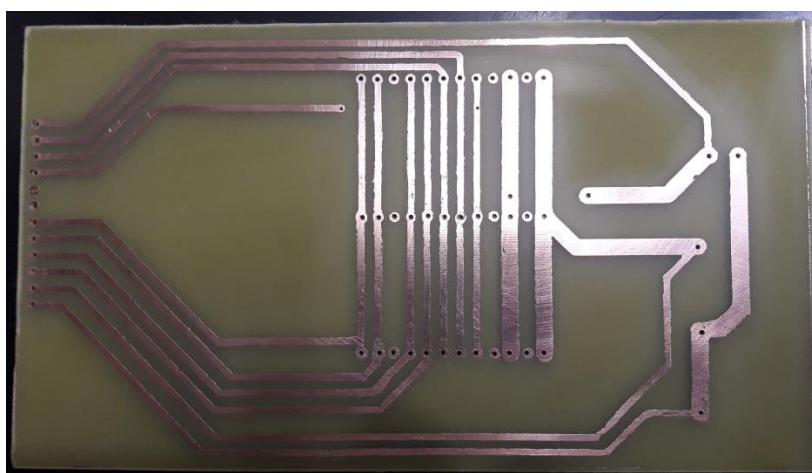


Fonte: Próprio autor

2.6.2 Placa Base

Na figura 29 pode-se observar a placa Base após a impressão e corrosão.

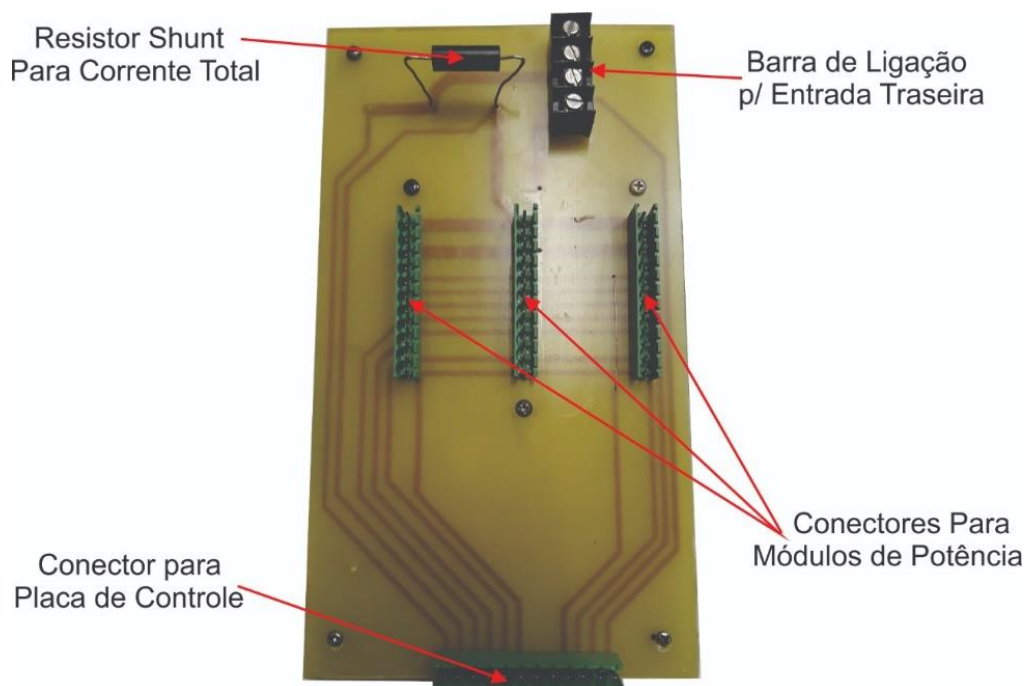
Figura 29 - Vista inferior da Placa Base



Fonte: Próprio autor

Na figura 30, pode-se observar a placa base montada. Esta placa comporta os conectores para os módulos de potência e para a placa de controle, O resistor Shunt Geral de 0.01R / 9W e a barra de ligação para a entrada traseira da Carga Eletrônica.

Figura 30 - Vista Superior da Placa Base Montada

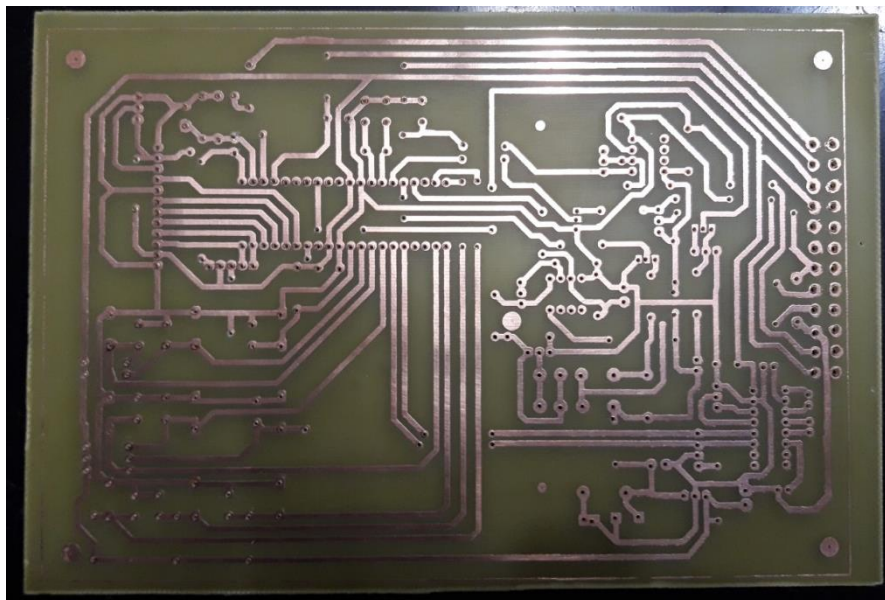


Fonte: Próprio autor

2.6.3 Placa de Controle

Na figura 31 pode-se observar a placa de Controle após a impressão e corrosão.

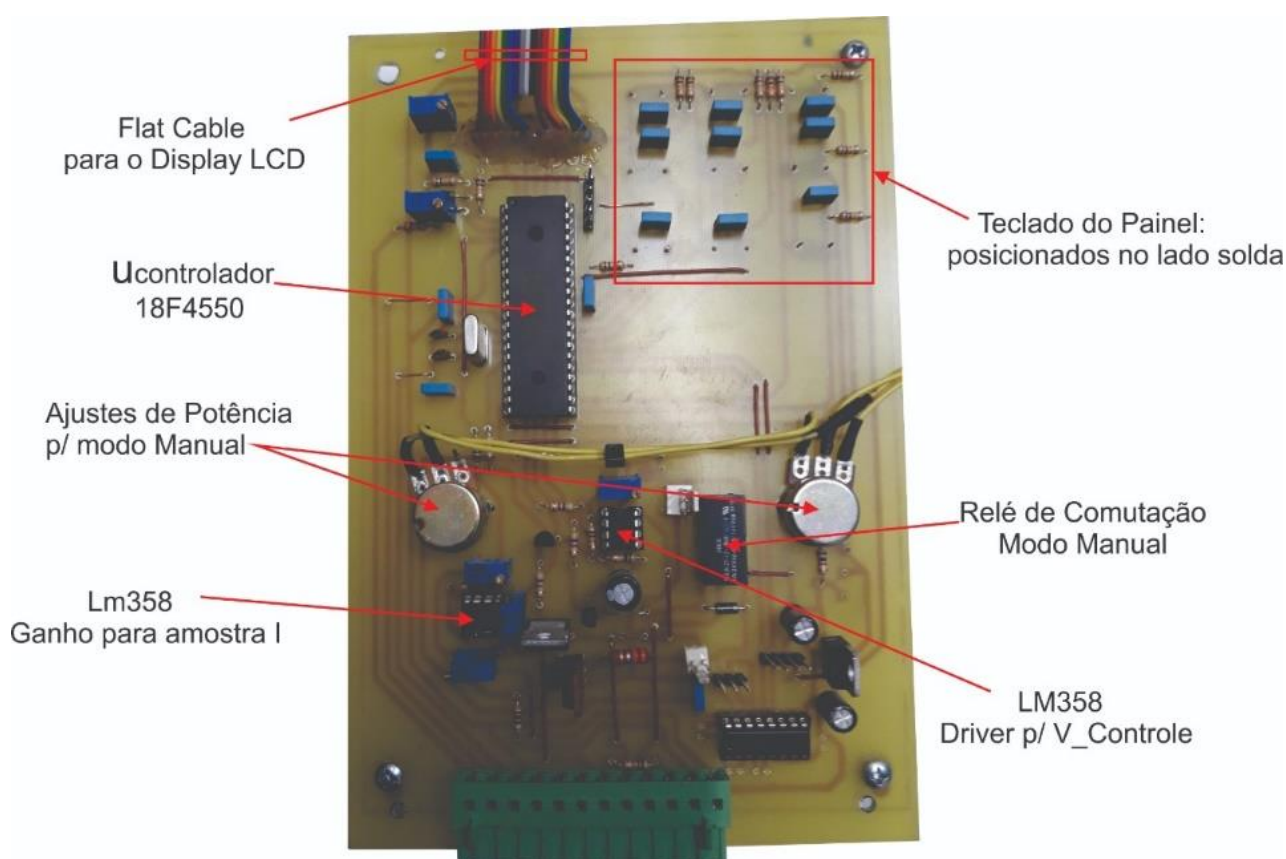
Figura 31 - Vista inferior da Placa de Controle



Fonte: Próprio autor

Na figura 32 pode-se observar a placa de controle já montada com todos os seus componentes. É aqui que é feita a gestão de todo o processo de funcionamento da Carga Eletrônica. Nesta placa se encontra localizado o microcontrolador 18F4550 que faz o gerenciamento e controle de todo o sistema, saída para o módulo display LCD através de flat cable, teclado para seleção e configuração dos modos de operação, driver de ganho de corrente para a tensão de controle, amplificador da tensão de referência para amostra de corrente, Potenciômetros para o ajuste manual da tensão de controle e relé de comutação para o modo de operação Manual.

Figura 32 - Placa de Controle montada



Fonte: Próprio autor

3 CONCLUSÃO

Após todo o processo aqui descrito, seguindo as etapas mencionadas, simulando o circuito proposto, fazendo as alterações necessárias e os ajustes pertinentes, adquirindo componentes adequados e de boa qualidade, montando o circuito com cuidado e manuseio adequado, o resultado obtido foi bastante satisfatório.

O Projeto Carga Eletrônica CC funcionou a contento, e as expectativas esperadas desde o início do processo até sua implementação final foram alcançadas.

Foram realizados testes com fontes de alimentação CC de 12V, 24V, e 36V nos cinco modos de operação e os resultados obtidos reafirmaram a eficácia do equipamento no que se refere ao objetivo a que se destina que é drenar corrente de forma controlado dos equipamentos a ela conectado para testes de suas especificações.

Em testes realizados com fontes de alimentação com ripple alto (acima de 5%), observou-se apenas uma variação nas leituras apresentadas no display, porém sem afetar a efetividade do ensaio que é drenar a corrente do equipamento submetendo-o a testes que comprovem as suas especificações de fábrica.

Em testes de descarga de baterias pode-se drenar a máxima corrente e verificar o aquecimento dos módulos de potência que se manteve estável à temperatura de 52°C.

Como indicação para trabalhos futuros, sugere-se que seja implementado um sistema supervisor onde, através do qual todo o processo de descarga da bateria sob testes ou durante a certificação das especificações dos equipamentos, possam ser gerados gráficos que descrevam todo o processo em tempo real e posteriormente relatórios finais dos testes realizados.

REFERÊNCIAS

ALBERT PAUL, Malvino. **Eletrônica 1**, Editora: Pearson; 4ª edição (1 janeiro 1996), Total de Páginas p.748

LUIZ VOLPIANO, Sérgio. **Eletrônica de potência (Eletroeletrônica)**, Editora: SENAI-SP; 1ª Edição (2 julho 2018), Total de Páginas p.281

PEREIRA, Fábio. **Microcontroladores PIC: Programação em C**, Editora: Érica; 7ª edição (2007) Total de Páginas p.360

W. HART, Daniel. **Eletrônica de Potência: Análise e Projetos de Circuitos**, Editora: AMGH; 1ª edição (1 janeiro 2011) Total de Páginas p.496

APÊNDICE A

A.1 PERT

Figura 33 - Atividades Hardware

ATIVIDADES (HARDWARE)	Tempo (dias)
AH = Projetar Circuito Eletrônico da Potência	5
BH= Projetar Circuito Eletônico do controle	10
CH= Simular Circuito Eletrônico da Potência no Proteus	3
DH = Simular Circuito Eletrônico do Controle no Proteus	3
EH = Comprar Componentes	8
FH = Validar Circuito de Potência na Matriz de Contatos	5
GH = Validar Circuito de Controle na Matriz de contatos	10
HH = Integrar Firmware ao Hardware	8
IH = Fazer Layout da PCI Potência	7
JH= Fazer Layout da PCI Controle	10
KH = Confeccionar a PCI de Potência	3
LH = Confeccionar a PCI de Controle	5
MH= Montar o circuito na PCI de potência	4
NH = Montar o Circuito na PCI Controle	6
OH = Fazer Ligações elétricas entre PCI's de Controle e Potência	3
PH = Validar/Testar Circuito Completo	3
QH = Integrar Mecânica ao Hardware	7
RH = Testar o Projeto	8

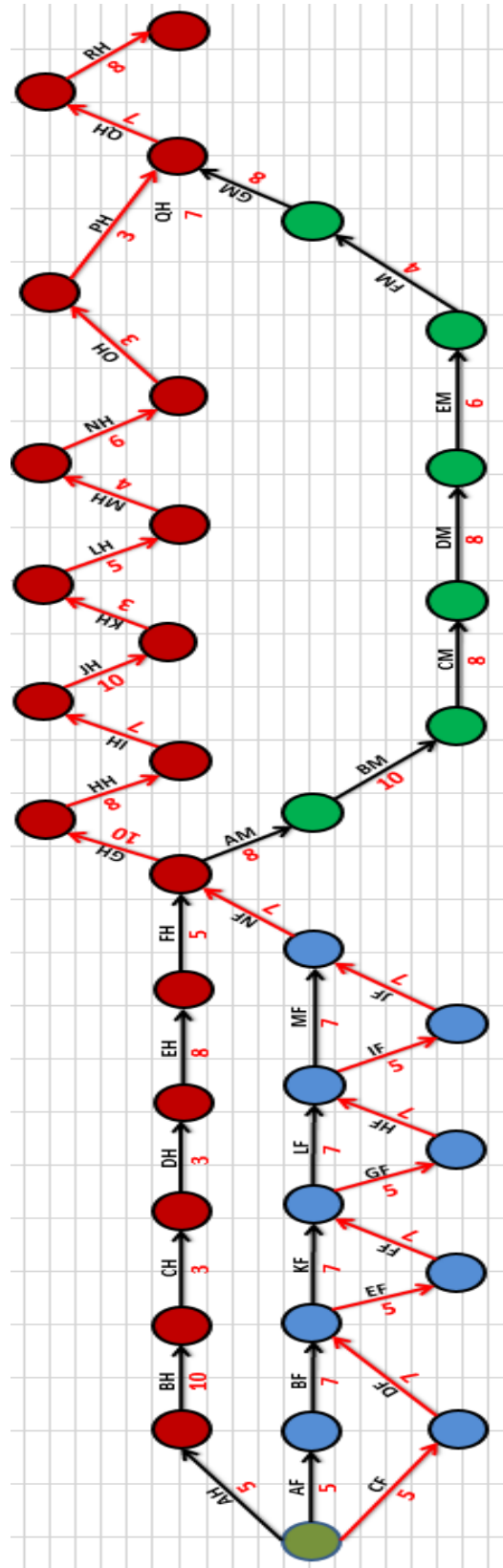
Figura 34 - Atividades Mecânica

ATIVIDADES (MECÂNICA)	Tempo (dias)
AM = Projetar Gabinete	8
BM = Projetar Painei do Gabinete	10
CM = Pesquisar desenvolvedores de gabinete	8
DM = Pesquisar Desenvolvedores do Painei	8
EM = Contratar Fabricante de gabinete e Painei	6
FM = Comprar Acessórios de Fixação do Hardware no Gabinete	4
GM = Integrar o Painei no Gabinete	8

Figura 35 - Atividades Firmware

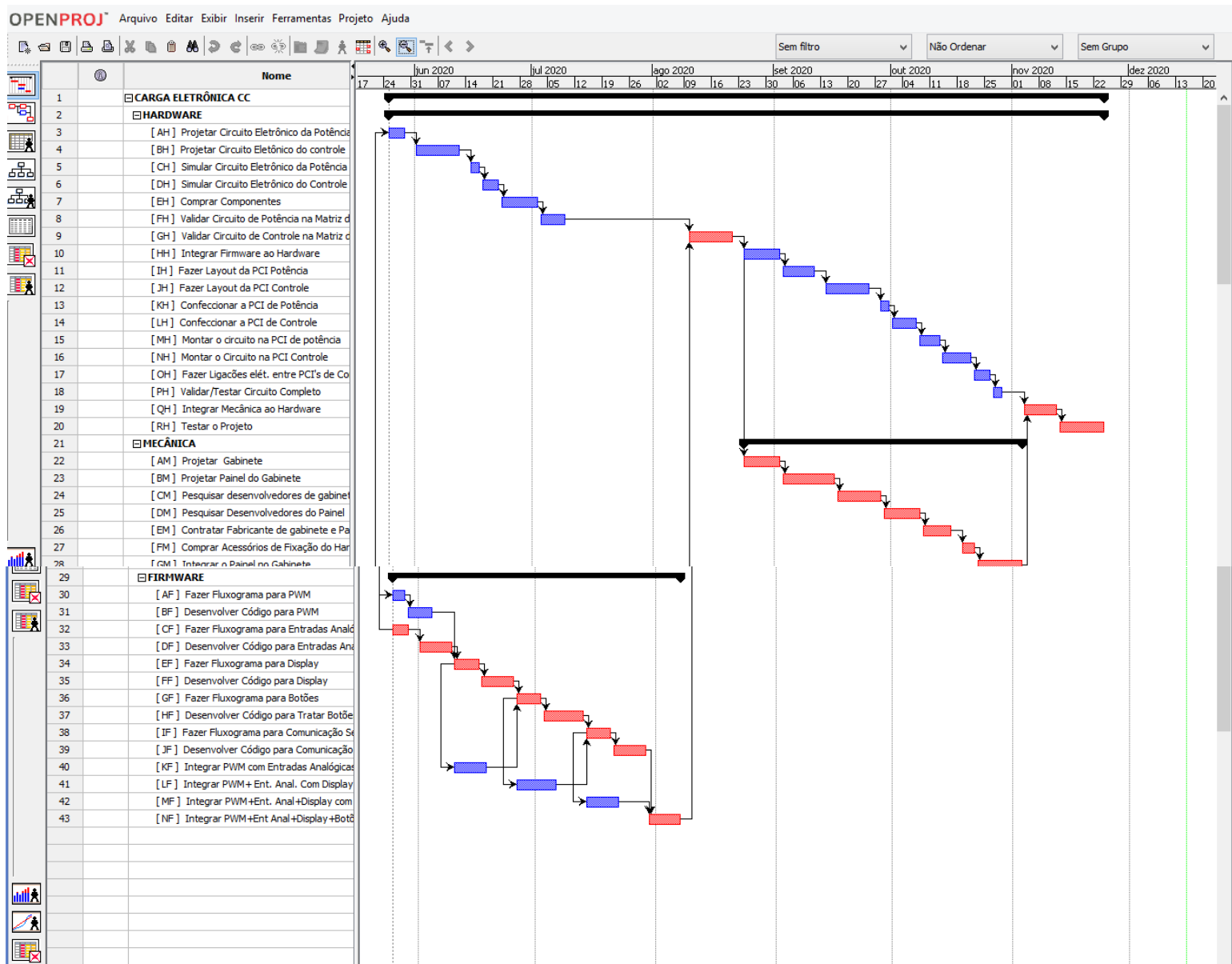
ATIVIDADES (FIRMWARE)	Tempo (dias)
AF = Fazer Fluxograma para PWM	5
BF = Desenvolver Código para PWM	7
CF = Fazer Fluxograma para Entradas Analógicas	5
DF = Desenvolver Código para Entradas Analógicas	7
EF = Fazer Fluxograma para Display	5
FF = Desenvolver Código para Display	7
GF = Fazer Fluxograma para Botões	5
HF = Desenvolver Código para Tratar Botões	7
IF = Fazer Fluxograma para Comunicação Serial	5
JF = Desenvolver Código para Comunicação serial	7
KF = Integrar PWM com Entradas Analógicas	7
LF = Integrar PWM+ Ent. Anal. Com Display	7
MF = Integrar PWM+Ent. Anal+Display com Botões	7
NF = Integrar PWM+Ent Anal+Display+Botões com Comun. Serial	7

Figura 36 - PERT Carga Eletrônica



A.2 GANT

Figura 37 - GANT CARGA ELETRÔNICA



A.3 EAP (ESTRUTURA ANALÍTICA DO PROJETO)

1 Carga Eletrônica CC

1.1 Hardware

1.1.1 Circuito

1.1.2 Dimensionamento

1.1.2.1 Atuadores

1.1.2.2 Controlador

1.1.2.3 Componentes

1.1.3 Fonte de Alimentação

1.2 Planejamento

1.2.1 Financeiro

1.2.1.1 Abertura de Conta Bancária

1.2.1.2 Arrecadação Mensal

1.2.2 Compras

1.2.2.1 Cotação

1.2.2.2 Entrega

1.2.3. Protótipo

1.2.4 Viabilidade

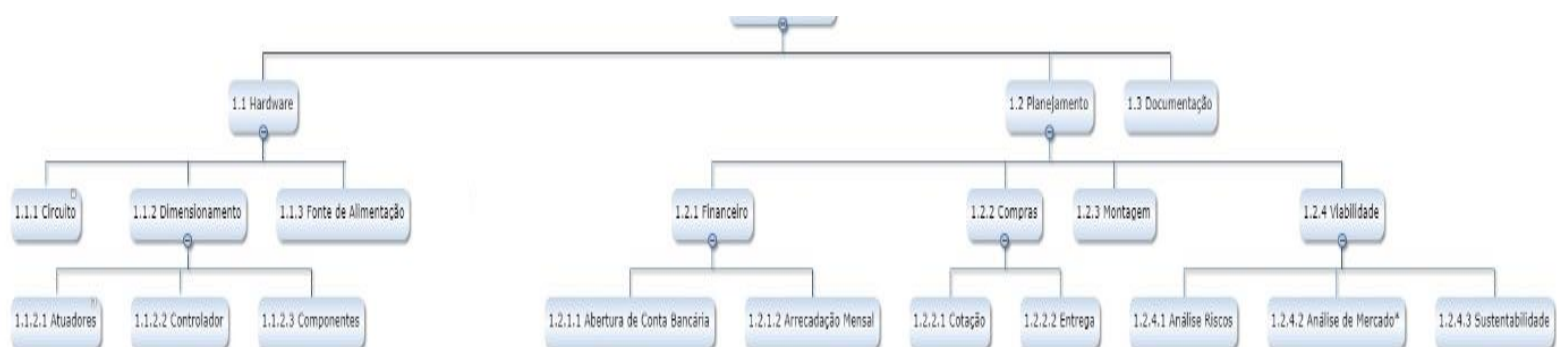
1.2.4.1 Análise Riscos

1.2.4.2 Análise de Mercado*

1.2.4.3 Sustentabilidade

1.3 Documentação

Figura 38 - EAP Carga Eletrônica



DICIONÁRIO DO EAP

1 Carga Eletrônica CC	Descrição: Projeto Conclusão de Curso
	Responsável: Davi, Joel, Leonardo, Fabrini
1.1 Hardware	Descrição: Hardware Carga Eletrônica CC, se trata de todos escopos construídos e testados, como Software, Dimensionamento, Circuitos, Atuadores etc
	Responsável: Davi
1.1.1 Circuito	Descrição: Após os dimensionamentos, a realização dos circuitos serão revisadas, testadas e calculadas. Este tópico apresenta o escopo de projeto dos esquemas elétricos.
	Responsável: Joel
1.1.2 Dimensionamento	Descrição: Análise, estudo e levantamento dos itens abaixo, tendo em paralelo os circuitos, o tópico apresenta, cada parte que será avaliada, para melhor funcionamento da Carga Eletrônica CC.
	Responsável: Joel
1.1.2.1 Atuadores	Descrição: Análise, estudo e levantamento, dos atuadores utilizados, na IHM do projeto, qual display será utilizado, botões, chaves seletores, potenciômetros de ajustes.
	Responsável: Joel

1.1.2.2 Controlador	Descrição: Dimensionamento do Microcontrolador utilizado no projeto. Analisar se o mesmo, atende todas as necessidades de entradas, saídas, compilação, programação.
	Responsável: Fabrini
1.1.2.3 Componentes	Descrição: Levantamento componentes utilizados, assim em conjunto com o dimensionamento dos circuitos, calculando e obtendo uma melhor estrutura para cada tópico, como fonte, software, controlador. Também avaliando melhor integração e versatilidade do projeto.
	Responsável: Davi
1.1.3 Fonte de Alimentação	Descrição: Levantamento alimentação do projeto, 110-220 Volts, alimentação do microcontrolador, display LCD, tendo em vista a capacidade máxima de 360W, quando utilizando uma carga em teste.
	Responsável: Leonardo
1.2 Planejamento	Descrição: Análise do Projeto no quesito Planejamento, com os tópicos: Financeiro, Compras, Protótipo.
	Responsável: Fabrini
1.2.1 Financeiro	Descrição: Análise Financeira do Projeto, elaborando em longo prazo, o que virá a ser gasto em projeto. Como a parte de fluxo de caixa, análise de compra e chegada dos componentes do projeto, em vista da data de chegada, custo-benefício e preço.
	Responsável: Joel
1.2.1.1 Abertura de Conta Bancária	Descrição: Elaboração de uma conta bancária para arrecadação de fundos do Projeto, onde será avaliado todos os custos em caixa que o projeto tem.
	Responsável: Davi

1.2.1.2 Arrecadação Mensal	Descrição: Será realizado aportes mensais para os investimentos feitos no projeto. Todo mês os integrantes, realizarão depósitos na conta do projeto.
	Responsável: Davi
1.2.2 Compras	Descrição: Análise de compra dos componentes que irão compor o projeto. Tendo em vista o dinheiro arrecadado que estará na conta bancária.
	Responsável: Leonardo
1.2.2.1 Cotação	Descrição: Análise de preços e custo benefício, dos componentes do projeto.
	Responsável: Leonardo
1.2.2.2 Entrega	Descrição: Data prevista do recebimento, tendo em vista, as datas que as atividades serão realizadas.
	Responsável: Fabrini
1.2.3 Protótipo	Descrição: Integração dos circuitos Hardware e Software, realização da primeira montagem, tendo em vista, revisões para um produto final sem correções futuras.
	Responsável: Joel
1.2.4 Viabilidade	Descrição: Levantamento de viabilidade do projeto, mensurando datas, e a conclusão do mesmo.
	Responsável: Joel
1.2.4.1 Análise Riscos	Descrição: Levantamento de riscos do projeto, como: datas de entrega de componentes, pandemia, datas de apresentações, testes realizados.
	Responsável: Joel
1.2.4.2 Análise de Mercado*	Descrição: Análise de mercado, será feita tendo em vista, o impacto que o projeto trará na indústria, custo-benefício do produto, prós, contras e melhorias do projeto.
	Responsável: Davi
1.2.4.3 Sustentabilidade	Descrição: Análise Sustentável, trará a visão que o projeto terá na sociedade comum. Tendo em vista

	melhorias, meio ambiente, mercado de trabalho, meio acadêmico.
	Responsável: Davi
1.3 Documentação	Descrição: Elaboração de Documentos do projeto, como: Canvas, EAP, Diagrama de Pert, Artigo Científico, Monografia.
	Responsável: Fabrini

APÊNDICE B

B.1 ROTINA PRINCIPAL

```
#include <Tcc_Pic18.h>
#include <config.h>

#int_TIMER1
void tempo (void)
{
    if(potencia>0)
    {
        if (cont ==2)
        {
            seg = seg+1;
            energia = energia + potencia;
            cont = 0;
        }

        cont++;
    }
    if (disp == 2)
    {
        display =1;
        disp=0;
    }
    disp++;

    set_timer1(3036);
}

void main()
{
    setup_timer_1(T1_INTERNAL |T1_DIV_BY_8);
    set_timer1(3036);
    enable_interrupts(INT_TIMER1);
    enable_interrupts(INT_RDA);
```

```

enable_interrupts(GLOBAL);

set_tris_a (0b00001111);
set_tris_b (0b00000011);
set_tris_c (0b10000000);
set_tris_d (0b11111111);
set_tris_e (0b00000000);
portc=0;

SETUP_ADC(ADC_CLOCK_INTERNAL);
SETUP_ADC_PORTS (ALL_ANALOG);

SETUP_CCP1(CCP_PWM);
SETUP_TIMER_2 (T2_DIV_BY_1,125,1);

SET_PWM1_DUTY(0);
delay_us (50);

lcd_ini();
leitura();

while(TRUE)
{
//*****//
//          Leitura da Tensão de Entrada          //
//*****//

for(r=0;r<100;r++)
{
set_adc_channel(0);
delay_us(5);
amostra_ve += READ_ADC();
}
amostra_ve=amostra_ve/100;
V_ent=(amostra_ve*(5.0/1023))*30.0;

//*****//
//          FALHA POR SOBRE VE          //
// Se tensão de entrada for superior a 136V, entra no modo //
// falha por SOBRE TENSÃO, onde o PWM é zerado interrompendo //
// o consumo e permanece assim até que o botão HOME seja pressionado. //
//*****//

if (V_ent >= 138.0)
{
modo = 8;
limpa_lcd();
while (modo==8)
{
SET_PWM1_DUTY (0);

```

```

    lcd_envia_byte (0, 0x80) ;
    printf (lcd_escreve, "  FALHA SOBRE VE  ");
    delay_ms (1) ;
    if(home==0)
    {
        ciclo=1;
        modo=0;
        valor1=valor2=valor3=valor4=valorTotal=0;
        limpa_lcd();
    }
}

//*****//
//          Leitura da Corrente de Entrada          //
//*****//
for(a=0;a<100;a++)
{
    set_adc_channel(1);
    delay_us(5);
    amostra_le += READ_ADC();
}
amostra_le = amostra_le/100;
I_ent=(amostra_le*(5.0/1023)*10.0);

//*****//
//          Cálculo da Potência Consumida          //
//*****//
potencia = (V_ent*I_ent);

//*****//
//          Leitura da Temperatura nos Módulos de Potência          //
//*****//

for(t=0;t<=100;t++)
{
    set_adc_channel(2);
    delay_us(5);
    amostra_temp += READ_ADC();
}
amostra_temp= amostra_temp/100;
temp =(amostra_temp*(5.0/1023)*100);

trataTemp ();    // Chama a função que irá tratar a temperatura lida.

//*****//
//          contador do tempo          //
//*****//
unsigned char uni_s, dez_s, uni_m, dez_m, uni_h, dez_h, cent_h;

```

```

if(seg > 59)
{
    min = min+1;
    seg = 0;

    if(min > 59)
    {
        hora = hora+1;
        min = 0;
    }
}
dez_s = seg/10;
uni_s = seg%10;
dez_m = min/10;
uni_m = min%10;
cent_h = hora/100;
dez_h = (hora%100)/10;
uni_h = (hora%100)%10;

//*****//
//                                //
//*****//

    enrg = (energia/3600.0);
    grava();

//*****//
//                                //
//*****//

    if(display == 1)
    {
        lcd_envia_byte (0, 0x94) ;
        printf (lcd_escreve, " %liW      ",potencia);
        lcd_envia_byte (0, 0xA1) ;
        printf (lcd_escreve, "%liWh    ",enrg);
        delay_ms (1) ;

        lcd_envia_byte (0, 0xC0) ;
        printf (lcd_escreve, " %.1fV      ", V_ent);
        lcd_envia_byte (0, 0xCD) ;
        printf (lcd_escreve, "%.1fA    ", I_ent);
        lcd_envia_byte (0, 0xD4) ;
        printf (lcd_escreve, "%li ",val_pwm);
        lcd_envia_byte (0, 0xE3) ;
        printf (lcd_escreve, "%.1fC",temp);
        delay_ms (1) ;
        display = 0;
    }
    lcd_envia_byte (0, 0xD9) ;

```

```

    printf (lcd_escreve,
"%i%i%i:i:%i%i:i:%i%i",cent_h,dez_h,uni_h,dez_m,uni_m,dez_s,uni_s);

    //*****//
    // Se a potência consumida for menor ou igual a 360W a Carga Eletrônica //
    // funciona normalmente executando todo o processo de funcionamento //
    //*****//

if(potencia<=370)
{
    if(potencia>=1)
    { output_high (PIN_E1); }
    else
    { output_low (PIN_E1); }

selectMode();

selectValue();

operationMode ();

    //*****//
    // Se durante o processo o botão home for pressionado a Carga interrompe //
    // o processo, entra no modo Seleção de Operação e fica aguardando. //
    //*****//

    if(home==0)
    {
        ciclo=1;
        modo=0;
        valor1=valor2=valor3=valor4=valorTotal=0;
        limpa_lcd();
    }
}

    //*****//
    //Caso a potência consumida seja superior a 360W o PWM é zerado,
interrompendo //
    // o consumo e mostra no display LCD que houve sobrepotência. permanece
neste //
    // estado por 1,5 segundo e sai novamente, ficando neste looping enquanto //
    // perdurar a sobrepotência. //
    //*****//

else
{
    val_pwm=0;
    SET_PWM1_DUTY(val_pwm);
    delay_us (50);

```

```

limpa_lcd();
lcd_envia_byte(0,0x80);
printf(lcd_escreve,"FALHA SOBREPOTENCIA ");
delay_ms(1);

lcd_envia_byte(0,0xC0);
printf(lcd_escreve,"    %liW    ",potencia);
delay_ms(1);

delay_ms(1500);
limpa_lcd ();
delay_ms(100);

if(modos==6)
{
    modos=5;
}
}
}
}

```

B.2 ROTINA GRAVAÇÃO

```

void grava(void)
{

data_H = energia >> 24;
data_HL = energia >> 16;
data_LH = energia >> 8;
data_L = energia % 16777216;
write_eeprom (5,data_H);
delay_ms(10);
write_eeprom (6,data_HL);
delay_ms(10);
write_eeprom (7,data_LH);
delay_ms(10);
write_eeprom(8,data_L);
delay_ms(10);
write_eeprom(13,seg);
delay_ms(10);
write_eeprom(15,min);
delay_ms(10);
write_eeprom(17,hora);
delay_ms(10);
}

```

B.3 ROTINA LEITURA

```
void leitura(void)
{

data_H = read_eeprom(5);
delay_ms(10);
data_HL = read_eeprom(6);
delay_ms(10);
data_LH = read_eeprom(7);
delay_ms(10);
data_L = read_eeprom(8);
delay_ms(10);
energia = (data_H * 16777216) + (data_HL * 65536) + (data_LH * 256) + data_L;
seg= read_eeprom(13);
delay_ms(10);
min= read_eeprom(15);
delay_ms(10);
hora = read_eeprom(17);
delay_ms(10);
}
```

B.4 ROTINA LCD

```
#ifndef lcd_enable
#define lcd_enable    pin_b3    // pino enable do LCD
#define lcd_rs        pin_b2    // pino rs do LCD
#define lcd_d5         pin_b5    // pino de dados d4 do LCD
#define lcd_d4         pin_b4    // pino de dados d5 do LCD
#define lcd_d7         pin_b7    // pino de dados d6 do LCD
#define lcd_d6         pin_b6    // pino de dados d7 do LCD
#endif

#define lcd_type 2
//byte CONST ini_lcd[4] = {0x20 | (lcd_type << 2), 0xf, 1, 6};
byte CONST ini_lcd[4] = {0x20 | (lcd_type << 2), 0x0c, 0,6};
```

Envio de NIBBLE para o lcd

```
void lcd_envia_nibble(byte dado )
```

```
// envia um dado de quatro bits para o display
{
    // coloca os quatro bits nas saidas
    output_bit(lcd_d4,bit_test(dado,0));
    output_bit(lcd_d5,bit_test(dado,1));
    output_bit(lcd_d6,bit_test(dado,2));
    output_bit(lcd_d7,bit_test(dado,3));
    // dá um pulso na linha enable
    output_high(lcd_enable);    //enable = 1
    output_low(lcd_enable);     //enable = 0
    // retorna ao ponto de chamada da função
}
```

Envio de um byte para o lcd

```
//Esta rotina irá enviar um byte ou um comando para o lcd conforme abaixo:
// -> endereco = 0 => a variavel dado será uma instrução
// -> endereco = 1 => a variavel dado será um caracter
```

```
void lcd_envia_byte( boolean endereco, byte dado )
{
    output_low(lcd_rs);
    output_bit(lcd_rs,endereco);    // seta o bit rs para instrução ou caracter
    delay_us(100);                  // aguarda 100 us
    output_low(lcd_enable);          // desativa a linha de enable
    lcd_envia_nibble(dado >> 4);     // envia a primeira parte do byte
    lcd_envia_nibble(dado & 0x0f);   // limpa a primeira parte e envia a segunda
}
```

```
//*****//
//          Envio de caracter para o display          //
//*****//
```

```
void lcd_escreve( char c)
// envia caractere para o display
{
    lcd_envia_byte(1,c);
}
```

```
//*****//
//          Função para limpar o LCD                    //
//*****//
// como esta função pode vir a ser muito utilizada, transformando-a numa
// função deixa o código HEX menor
```

```
void limpa_lcd()
{
    lcd_envia_byte(0,0x01);    // envia a instrução para limpar o lcd
    delay_ms(5);//10ms 16/04/10
```



```
}
```

Inicialização do display

```
void lcd_ini()
// rotina de inicialização do display
{
    byte conta;
    output_low(lcd_d4);    //garante que o pino d4 está em 0
    output_low(lcd_d5);    //garante que o pino d5 está em 0
    output_low(lcd_d6);    //garante que o pino d6 está em 0
    output_low(lcd_d7);    //garante que o pino d7 está em 0
    output_low(lcd_rs);    //garante que o pino rs está em 0
    //output_high(lcd_rw);
    output_low(lcd_enable); //garante que o pino enable está em 0
    delay_ms(16);

    //envia uma sequencia de 3 vezes 0x03 e depois 0x02 para configura o modulo
    //para modo 4 bits
    for(conta=0;conta<=3;++conta)
    {
        lcd_envia_nibble(0x03); // Envia comando para inicializar o display
        delay_ms(4);           // Aguarda 5ms para estabilizar o LCD
    }
    lcd_envia_nibble(0x02);

    // CURSOR HOME - Envia comando para zerar o contador de caracteres e
    //retornar à posição inicial (0x80)
    for(conta=0;conta<=3;++conta)

        lcd_envia_byte(0,INI_LCD[conta]);
}
```

B.5 ROTINA MODO DE OPERAÇÃO

```
void operationMode ()
{
    SWITCH (modo)
    {
        //*****//
        //                Modo Potência Constante                //
        //*****//
        CASE 1:
        float potSel; // criada a variável para armazenar o valor da pot. escolhida
        output_low (PIN_C1); // mantém relé man/aut desl (Modo automático)
```

```

    potSel = valorTotal; // o valor escolhido de potência é transferido de
"valorTotal" para "potSel"

//*****//
//          Se potência FOR menor ou igual a 50w          //
// a função Select é acionada e apenas uma célula de potência fica ativa, //
// as demais ficam desabilitadas.          //
//*****//

IF (potSel <= 50)
{
    output_high (PIN_C0);
    potSel = potSel;

    // fórmula para determinar o valor do PWM correspondente á potência
    escolhida.
    val_pwm = ( (potSel / V_ent) / 5.0) * 512.0;
}

//*****//
//          Se potência for maior que 50W ou menor/igual á 360w          //
// A função Select é desligada e todos os módulos de potência ficam ativos. //
// O valor da potência escolhido é dividido em 6 para ser distribuído cada //
// parte para uma célula de potência          //
//*****//

IF (potSel > 50 && potSel <= 360)
{
    output_low (PIN_C0);
    potSel = potSel / 6.0;

    // Fórmula para determinar o valor do PWM correspondente á potência
    escolhida.
    // A constante 1.07 é para compensar a perda de tensão e corrigir a potência
    val_pwm = (( (potSel / V_ent) / 5.0) * 512.0)*1.07;
}

// limita o valor máximo do PWM em 510.
IF (val_pwm > 510) val_pwm = 510;

//insere o valor de PWM obtido na fórmula, na função PWM.
SET_PWM1_DUTY (val_pwm);
delay_us (50);

// escreve no display LCD o modo de operação.
lcd_envia_byte (0, 0x81);
printf (lcd_escreve, "P_CONSTANTE: %.1fW", valorTotal);
delay_ms (1) ;

//*****//

```

```

//          FALHA POR SUB VE          //
//  Se tensão de entrada for inferior a 10,8V, entra no modo      //
//  falha por SUB TENSÃO, onde o PWM é zerado interrompendo      //
//  o consumo e o display LCD indica a falha. Permanece assim até que  //
//  o botão HOME seja pressionado.          //
//*****//

if (V_ent <= 10.8)
{
    modo = 7;
    limpa_lcd();
    while (modo==7)
    {
        SET_PWM1_DUTY (0);
        lcd_envia_byte (0, 0x80) ;
        printf (lcd_escreve, "  FALHA SUB VE  ");
        delay_ms (1) ;
        if(home==0)
        {
            ciclo=1;
            modo=0;
            valor1=valor2=valor3=valor4=valorTotal=0;
            limpa_lcd();
        }
    }
}

BREAK;

//*****//
//          Modo Tensão Constante          //
//*****//
CASE 2:

output_low (PIN_C1);
output_low (PIN_C0);
vSel = valorTotal;
vSP = (vSel / 30.0) / (5.0 / 1023.0);
    vSel1 = vSel * 0.99;
    vSel2 = vSel * 1.01;

IF (vSel < V_ent)
{
    val_pwm = val_pwm + 2;

    IF (val_pwm > 510)
    {
        val_pwm = 510;
    }
}

```

```

    IF (vSel > V_ent)
    {
        val_pwm = val_pwm - 2;

        IF (val_pwm < 2)
        {
            val_pwm = 2;
        }
    }

    SET_PWM1_DUTY (val_pwm);
    delay_us (50);

    lcd_envia_byte (0, 0x81);
    printf (lcd_escreve, "V_CONSTANTE: %.1fV", vSel);
    delay_ms (1) ;

    IF ( V_ent > vSel1 && V_ent< vSel2)
    {
        val_pwm = val_pwm;
        modo = 21;
        limpa_lcd();
    }

    BREAK;

CASE 21:
    vSel1 = vSel * 0.95;
    vSel2 = vSel * 1.05;

    SET_PWM1_DUTY (val_pwm);          // o duty cicle DO PWM é determinado
    pelo valor da variável v_controle
    delay_us (50);

    lcd_envia_byte (0, 0x81) ;
    printf (lcd_escreve, "V_CONSTANTE: %.1fV", vSel);
    delay_ms (1) ;

    IF (V_ent <= vSel1||V_ent >= vSel2)
    {
        modo = 2;
    }

    BREAK;

//*****//
//                      Modo Corrente Constante                      //
//*****//
CASE 3:

```

```

FLOAT lsel;
output_low (PIN_C1);
lsel = valorTotal;

IF (lsel <= 5)
{
    output_high (PIN_C0);
    lsel = lsel;
    val_pwm = (lsel / 5.0) * 512.0;
}

IF (lsel > 5 && lsel <= 30)
{
    output_low (PIN_C0);
    lsel = lsel / 6.0;
    val_pwm = ((lsel / 5.0) * 512.0)*1.09;
}

SET_PWM1_DUTY (val_pwm);          // o duty cycle DO PWM é determinado
pelo valor da variável v_controle
delay_us (50);

lcd_envia_byte (0, 0x81) ;
printf (lcd_escreve, "I_CONSTANTE: %.1fA", valorTotal);
delay_ms (1) ;

BREAK;

//*****//
//                Modo Resistência Constante                //
//*****//
CASE 4:

FLOAT R_desejada,Rmin;
output_low (PIN_C1);  // desliga rele de comutação, desabilitando o ajuste
manual através de potenc.

R_desejada = valorTotal;

Rmin = (V_ent/R_desejada);

IF ( Rmin > 0 && Rmin <= 4 )
{
    output_high (PIN_C0);    // liga tensão + 12Vsw deixando apenas uma célula
de potência ativa.

```

// fórmula para determinar o valor do PWM correspondente á resistência escolhida.

```
val_pwm = (V_ent / R_desejada) / 5.0 * 512.0;
```

```
}
```

```
IF ( Rmin > 4 && Rmin <= 30 )
```

```
{
```

output_low (PIN_C0); // desliga tensão + 12Vsw deixando todas as célula de potência ativas.

// fórmula para determinar o valor do PWM correspondente á resistência escolhida.

```
val_pwm = (V_ent / R_desejada) / 5.0 * 512.0;
```

```
val_pwm = (val_pwm/6.0)*1.07;
```

```
}
```

```
IF (val_pwm > 510)
```

```
{
```

```
val_pwm = 510;
```

```
}
```

SET_PWM1_DUTY (val_pwm); // o duty cycle DO PWM é determinado pelo valor da variável v_controle
delay_us (50);

```
lcd_envia_byte (0, 0x81) ;
```

```
printf (lcd_escreve, "R_CONSTANTE: %.1fR", R_desejada);
```

```
delay_ms (1) ;
```

```
BREAK;
```

```
//*****//
```

```
// Modos Manual //
```

```
//*****//
```

```
CASE 5:
```

output_low (PIN_C0); // desliga tensão + 12Vsw liberando todos os módulos de potência

output_high (PIN_C1); // liga rele de comutação para o ajuste manual através de potenc.

```
// val_pwm = (82.0 / amostra_ve * 1024 / 2);
```

```
val_pwm=((60.0/V_ent)/5.0)*512.0;
```

```
if (val_pwm>512) { val_pwm = 512;}
```

SET_PWM1_DUTY (val_pwm); // o duty cycle DO PWM é determinado pelo valor da variável v_controle

```

delay_us (50);

//-----
amostra_ve1 = amostra_ve * 0.9;
amostra_ve2 = amostra_ve * 1.1;
modo = 6;

BREAK;

CASE 6 :
output_high (PIN_C1);
lcd_envia_byte (0, 0x84) ;
printf (lcd_escreve, "MOD0 MANUAL");
delay_ms (1) ;

IF (amostra_ve <= amostra_ve1||amostra_ve >= amostra_ve2)
{
    modo = 5;
}

BREAK;
}
}

```

B.6 ROTINA SELEÇÃO DE MODO

```

void selectMode()
{
if (letra==9)
{
limpa_lcd();
letra = 0;
}
while(ciclo==1)
{
    SET_PWM1_DUTY(0);
    delay_us (50);
    val_pwm = 0;
    potencia = 0;

    lcd_envia_byte(0,0x80);
    printf(lcd_escreve,"SELECIONE O MODO:  ");
    delay_ms(1);

    output_low(PIN_E0);
    output_low (PIN_E1);

    if(bot_mod0==0)

```

```

{
    modo++;
    delay_ms(300);
}

if(bot_desce == 0)
{
    limpa = 1;
    limpa_lcd();
    while(limpa == 1)
    {
        lcd_envia_byte(0,0xC0);
        printf(lcd_escreve," APAGAR MEMORIA ");
        delay_ms(1);

        if (enter == 0)
        {
            lcd_envia_byte(0,0xC0);
            printf(lcd_escreve," MEMORIA LIMPA ");
            delay_ms(3000);
            seg = 00;
            min = 00;
            hora = 00;
            energia = 0;
            limpa = 0;
            limpa_lcd();
        }
        if (home == 0)
        {
            limpa=0;
            limpa_lcd();
        }
    }
}

if(modo==1)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"POTENCIA CONSTANTE ");
    delay_ms(1);
}

if(modo==2)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"TENSAO CONSTANTE ");
    delay_ms(1);
}

if(modo==3)

```



```

{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"CORRENTE CONSTANTE ");
    delay_ms(1);
}

if(modos==4)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"RESIST CONSTANTE ");
    delay_ms(1);
}

if(modos==5)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"MODO MANUAL ");
    delay_ms(1);
}

if(modos>5)
{
    modos=1;
}
if(modos==1||modos==2||modos==3||modos==4)
{
    if(enter==0)
    {
        ciclo=2;
        delay_ms(300);
        limpa_lcd();
    }
}

if(modos==5)
{
    if(enter==0)
    {
        ciclo=0;

        limpa_lcd();
    }
}

}
}

```

B.7 ROTINA SELEÇÃO VALOR

```

void selectMode()
{
  if (letra==9)
  {
    limpa_lcd();
    letra = 0;
  }
  while(ciclo==1)
  {
    SET_PWM1_DUTY(0);
    delay_us (50);
    val_pwm = 0;
    potencia = 0;

    lcd_envia_byte(0,0x80);
    printf(lcd_escreve,"SELECIONE O MODO:  ");
    delay_ms(1);

    output_low(PIN_E0);
    output_low (PIN_E1);

    if(bot_modos==0)
    {
      modo++;
      delay_ms(300);
    }

    if(bot_desce == 0)
    {
      limpa = 1;
      limpa_lcd();
      while(limpa == 1)
      {
        lcd_envia_byte(0,0xC0);
        printf(lcd_escreve,"  APAGAR MEMORIA ");
        delay_ms(1);
      }

      if (enter == 0)
      {
        lcd_envia_byte(0,0xC0);
        printf(lcd_escreve,"  MEMORIA LIMPA ");
        delay_ms(3000);
        seg = 00;
        min = 00;
        hora = 00;
        energia = 0;
        limpa = 0;
        limpa_lcd();
      }
    }
  }
}

```

```

    if (home == 0)
    {
        limpa=0;
        limpa_lcd();
    }
}

if(modos==1)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"POTENCIA CONSTANTE ");
    delay_ms(1);
}

if(modos==2)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"TENSÃO CONSTANTE ");
    delay_ms(1);
}

if(modos==3)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"CORRENTE CONSTANTE ");
    delay_ms(1);
}

if(modos==4)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"RESIST CONSTANTE ");
    delay_ms(1);
}

if(modos==5)
{
    lcd_envia_byte(0,0xC0);
    printf(lcd_escreve,"MODO MANUAL ");
    delay_ms(1);
}

if(modos>5)
{
    modos=1;
}
if(modos==1||modos==2||modos==3||modos==4)
{
    if(enter==0)

```

```
{
  ciclo=2;
  delay_ms(300);
  limpa_lcd();
}

if(modo==5)
{
  if(enter==0)
  {
    ciclo=0;

    limpa_lcd();
  }
}

}
```