

UNIVERSIDADE FEDERAL DO PARANÁ

DANIEL SILVEIRA ROCHA

RODRIGO FORMIGHIERI

AUTOSIMOD - SIMULADOR MODULAR DIGITAL PARA ENSAIOS DO
AUTOMATISMO DO SISTEMA DE SERVIÇOS AUXILIARES DE UMA SUBESTAÇÃO

CURITIBA

2019

DANIEL SILVEIRA ROCHA
RODRIGO FORMIGHIERI

AUTOSIMOD - SIMULADOR MODULAR DIGITAL PARA ENSAIOS DO
AUTOMATISMO DO SISTEMA DE SERVIÇOS AUXILIARES DE UMA SUBESTAÇÃO

Trabalho de conclusão de curso de graduação, apresentado ao Curso de Engenharia Elétrica com Ênfase em Sistemas Eletrônicos Embarcados da Universidade Federal do Paraná como requisito à obtenção do grau de Engenheiro Eletricista, Setor de Tecnologia da Universidade Federal do Paraná.

Orientador: Prof. Dr. Marcos Vinicio Haas Rambo

CURITIBA
2019

TERMO DE APROVAÇÃO

**DANIEL SILVEIRA ROCHA
RODRIGO FORMIGHIERI**

**AUTOSIMOD - SIMULADOR MODULAR DIGITAL PARA ENSAIOS DO
AUTOMATISMO DO SISTEMA DE SERVIÇOS AUXILIARES DE UMA
SUBESTAÇÃO**

Trabalho de conclusão de curso, aprovado como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica, Setor de Tecnologia, Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientador: Prof. Dr. Marcos V. H. Rambo
Departamento de Engenharia Elétrica
UFPR

Prof. M.Sc Bruno Pohlott Ricobom
Departamento de Engenharia Elétrica
UFPR

Prof. Armando Heilmann
Departamento de Engenharia Elétrica
UFPR

Curitiba, 25 de Junho de 2019.

Dedicamos este trabalho às nossas famílias que sempre nos apoiaram, acreditaram na nossa capacidade e nos incentivam a dar o nosso melhor todos os dias.

AGRADECIMENTOS

Agradecemos aos nossos familiares, por sempre estarem do nosso lado e nos incentivarem a seguir em frente todos os dias.

A todos os professores do Departamento de Engenharia Elétrica da UFPR, em especial ao Prof. Dr. Marcos Vinicio Haas Rambo por nos orientar neste trabalho de conclusão de curso, nos direcionando ao caminho do aprendizado e na busca pelo conhecimento que se tornou base para a conclusão deste projeto.

Enfim, a todos àqueles que, durante o período da graduação, contribuíram para que alcançássemos os nossos objetivos na Universidade Federal do Paraná.

RESUMO

O presente projeto tem por finalidade o desenvolvimento de um protótipo modular de verificação e simulação, em tempo real, do sistema de serviços auxiliares de uma subestação de energia elétrica. Este sistema é composto por dispositivos que operam e controlam todos os equipamentos principais em uma subestação. Para tanto é necessária a utilização de um Controlador de Automação em Tempo Real (RTAC), para gerir, através de lógicas embarcadas, o funcionamento do processo. Para garantir a melhor confiabilidade dos recursos, o protótipo tem por desafio comandar e verificar estados lógicos, afim de validar se a sequência de processos a serem realizados pelo controlador está em acordo com as normas de funcionamento da Companhia Paranaense de Energia (Copel), assegurando o perfeito funcionamento de todos os equipamentos da subestação. Para efetiva conclusão do trabalho, houve a divisão da metodologia em algumas fases. Inicialmente foi realizada uma pesquisa bibliográfica, no intuito de consolidar os estudos relacionados à subestações, serviços auxiliares, e equipamentos do sistema elétrico de potência. Visto isso, se fez necessário o estudo de dispositivos eletrônicos capazes de transmitir sinais lógicos digitais através dos sinais emitidos pelo RTAC. Assim desenvolveu-se um hardware composto por módulos de comando e de estado. Por último foi programado o software embarcado a um Raspberry Pi Model B com capacidade de interação gráfica com o usuário, e verificação do funcionamento por tela TFT *touchscreen* associada ao computador. Todos os testes de validação do protótipo e análise dos resultados foram realizados em simulações de bancada na Copel, junto aos engenheiros e técnicos da empresa que compõem o corpo funcional da mesma.

Palavras-chaves: Subestação, serviços auxiliares, RTAC, Copel

ABSTRACT

The purpose of this project is the development of a modular prototype that verifies and simulates, in real time, the auxiliary services system of an electrical substation. This system is composed of devices that operate and control all the main equipment in a substation. Therefore, it is necessary to use a Real Time Automation Controller (RTAC), to manage, through embedded logics, the operation of the process. In order to guarantee the best reliability of the resources, the prototype has the challenge of commanding and verifying logical states, with the purpose of validating if the sequence of processes to be performed by the controller is in accordance with Copel's (Paranaense Energy Company) rules of operation, ensuring the perfect functioning of all substation equipment. For the effective conclusion of this work, the methodology was divided in some steps. Initially a bibliographical research was carried out in order to consolidate the studies related to substations, auxiliary services, and power system equipment. Thereby, it was necessary to study the capability of electronic devices transmitting digital logic signals through the ones emitted by the RTAC. Thus it was developed a hardware composed by modules of command and state. Finally, an embedded software was programmed to a Raspberry Pi Model B with graphical user interaction capability, and verification of the operation through a TFT *touchscreen* associated to this computer. All the prototype validation tests and results analysis were performed in Copel's simulations bench, with engineers and technicians of the company that compose the functional body of it.

Key-words: Substation, auxiliary services, RTAC, Copel

LISTA DE ILUSTRAÇÕES

FIGURA 2.1 – SUBESTAÇÃO CURITIBA NORTE - CTN	20
FIGURA 2.2 – DIAGRAMA DE ALIMENTAÇÃO DOS SA - SACA - CTN	22
FIGURA 2.3 – DIAGRAMA UNIFILAR DE ALIMENTAÇÃO - SACA - CTN	23
FIGURA 2.4 – DIAGRAMA UNIFILAR DE ALIMENTAÇÃO - SACC - CTN	23
FIGURA 2.5 – DIAGRAMA DE ALIMENTAÇÃO - SACC - CTN	24
FIGURA 2.6 – RTAC SEL-2243 DA SCHWEITZER NA SUBESTAÇÃO CTN	26
FIGURA 2.7 – ALARMES E ESTADOS DIGITAIS EM TEMPO REAL DOS EQUIPAMENTOS DA SUBESTAÇÃO CTN. DESTAQUE PARA ESTADOS DOS DISJUNTORES DA ALIMENTAÇÃO DOS SERVIÇOS AUXILIARES EM CC	26
FIGURA 2.8 – CENÁRIO HIPOTÉTICO DE FALHA NO TRANSFORMADOR PRINCIPAL DE SERVIÇOS AUXILIARES E SUBSEQUENTES ATUAÇÕES DO RTAC .	27
FIGURA 2.9 – ESQUEMA BÁSICO DE COMUNICAÇÃO EM PROTOCOLO I2C	28
FIGURA 2.10 – BLOCOS DE 8 BITS DE ESCRITA E LEITURA DO PROTOCOLO I2C .	29
FIGURA 2.11 – COMPARATIVO ENTRE AS VERSÕES DO RASPBERRY PI MAIS ATUAIS	30
FIGURA 2.12 – MODELO RASPBERRY PI B+	31
FIGURA 3.1 – DIAGRAMA ELEMENTAR DE CONEXÕES INTERNAS DO SIMULADOR E CONEXÕES EXTERNAS COM O RTAC	34
FIGURA 3.2 – MÓDULO DE ESTADOS UTILIZADO ATUALMENTE PELA COPEL	36
FIGURA 3.3 – CIRCUITO ESQUEMÁTICO DE APENAS 1 DOS 16 MÓDULOS DE ENTRADA DA NOVA INTERFACE DE ESTADOS PROPOSTA	37
FIGURA 3.4 – CIRCUITO ESQUEMÁTICO DA INTERFACE DE COMANDOS UTILIZADA PELA COPEL ATUALMENTE	39
FIGURA 3.5 – CIRCUITO ESQUEMÁTICO DO NOVO MÓDULO DE COMANDO	40
FIGURA 3.6 – ENDEREÇAMENTO DOS MCP23017 PARA IDENTIFICAÇÃO DOS COMANDOS DO PROTOCOLO I2C	42
FIGURA 3.7 – CIRCUITO DE ENDEREÇAMENTO DO MÓDULO DE ESTADOS	43
FIGURA 3.8 – CIRCUITO DE ENDEREÇAMENTO DO MÓDULO DE COMANDOS	43

FIGURA 3.9 – DIAGRAMA DE CASO DE USO COMPLETO DO SOFTWARE SIMULADOR	45
FIGURA 3.10 – TELAS DE INÍCIO E CARREGAMENTO DE ARQUIVO DE CONFIGURAÇÃO DO SOFTWARE DE SIMULAÇÃO	46
FIGURA 3.11 – EXEMPLO DE ARQUIVO DE CONFIGURAÇÃO JSON QUE PODE SER CARREGADO NO SOFTWARE DE SIMULAÇÃO, CONTENDO INFORMAÇÃO A RESPEITO DOS EQUIPAMENTOS E PORTAS UTILIZADAS	47
FIGURA 3.12 – TELA DE CADASTRO DE EQUIPAMENTOS DA SE QUE SERÃO SIMULADOS. O USUÁRIO DEVE INSERIR O NOME DE CADA EQUIPAMENTO	48
FIGURA 3.13 – TELA DE MAPEAMENTO DAS PORTAS DO SIMULADOR AOS EQUIPAMENTOS PREVIAMENTE CADASTRADOS	49
FIGURA 3.14 – TELA PRINCIPAL DE SIMULAÇÃO DO SOFTWARE	49
FIGURA 3.15 – TELA DE CRIAÇÃO DE SCRIPTS. EM EVIDÊNCIA A INSERÇÃO DE UMA INSTRUÇÃO DE DESATIVAMENTO DE ALARME “TAP LOW DIESEL” DO EQUIPAMENTO "GMG"	51
FIGURA 3.16 – TELA DE CRIAÇÃO DE SCRIPTS CONTENDO UM SCRIPT EXEMPLO COM 6 INSTRUÇÕES. À ESQUERDA OS BOTÕES QUE ADICIONAM INSTRUÇÕES, À DIREITA AS INSTRUÇÕES JÁ ADICIONADAS	52
FIGURA 3.17 – CONFIGURAÇÃO SALVA DO SCRIPT DA FIGURA 3.17 EM FORMATO JSON	53
FIGURA 3.18 – TELA PRINCIPAL DE SIMULAÇÃO COM EVIDENCIAÇÃO DO MENU DE EXECUÇÃO DE SCRIPTS	54
FIGURA 3.19 – TELA DE EXIBIÇÃO GRÁFICA ESPECIALMENTE DESENVOLVIDA PARA SIMULAÇÃO DO SACA DA CTN. ENTRADAS E SAÍDAS PODEM SER ATRELADAS AOS ELEMENTOS GRÁFICOS	54
FIGURA 3.20 – TELA DE EXIBIÇÃO GRÁFICA ESPECIALMENTE DESENVOLVIDA PARA SIMULAÇÃO DO SACC DA CTN, SENDO EVIDENCIADO O ATRELAMENTO DA SAÍDA DE SOBRETENSÃO DO DISJUNTOR “DG3” AO ELEMENTO GRÁFICO QUE O REPRESENTA	55

FIGURA 3.21 – INTERAÇÃO ENTRE SOFTWARE E HARDWARE. O PROTOCOLO I2C INTERMEDIA OS COMANDOS RECEBIDOS PELO RTAC E OS ESTADOS ENVIADOS A ELE ATRAVÉS DO SOFTWARE	56
FIGURA 3.22 – INTERPRETAÇÃO DAS INTERRUPÇÕES ACIONADAS PELO CI MCP23017 NO MÓDULO DE COMANDOS, AO RECEBER UM EVENTO DE COMUTAÇÃO DE PORTA VINDA DO RTAC EM FORMA DE PULSO NA PORTA 28	57
FIGURA 3.23 – ROTINA DE COMUTAÇÃO DE PORTAS DE SAÍDA DO ME. A FUNÇÃO “WRITE-OUTPUT” RECEBE COMO ARGUMENTOS A PORTA QUE SERÁ COMUTADA E O NOVO ESTADO DA MESMA.	58
FIGURA 4.1 – PLACA PCB DO MÓDULO EXPANSOR DE PORTAS	60
FIGURA 4.2 – PLACA PCB DO MÓDULO EXPANSOR DE PORTAS JÁ COM OS COMPONENTES SOLDADOS. EM EVIDÊNCIA OS MÓDULOS DE ESTADOS E COMANDOS, QUE CONTÉM AS PORTAS DIGITAIS A SEREM CONECTADOS NO RTAC	60
FIGURA 4.3 – TESTE DE BANCADA DAS RESPOSTAS DO RTAC JUNTO AO HARDWARE DO PROTÓTIPO	61
FIGURA 4.4 – MONTAGEM FÍSICA DO TESTE DE BANCADA REALIZADO COM TODAS AS 16 PORTAS DE SAÍDA (DE 4 EM 4). EM EVIDÊNCIA OS COMPONENTES ENVOLVIDOS NO ENSAIO	62
FIGURA 4.5 – TESTE DE BANCADA REALIZADO (ANÁLISE DO FUNCIONAMENTO DO PROTOCOLO I2C NA COMUNICAÇÃO COM O ME). AS PORTAS 22, 23, 24 E 25 DO ME CONTROLAM O ASCENDIMENTO DOS LEDs 1, 2, 3 E 4 RESPECTIVAMENTE.	63
FIGURA 4.6 – SCRIPT DE TESTE DO ME	64
FIGURA 4.7 – RELATÓRIO DO TESTE DO ME. OS LEDs ASCENDEM SEQUENCIALMENTE EM INTERVALOS DE 1 SEGUNDO E POR FIM TODOS SÃO DESLIGADOS AO MESMO TEMPO APÓS 10 SEGUNDOS CONFORME INSTRUÇÕES DO SCRIPT	64
FIGURA 4.8 – MAPEAMENTO DAS 16 PORTAS PARA OS ENSAIOS E TESTES DO MC	65

FIGURA 4.9 – RELATÓRIO DE TESTE DO MC REALIZADO. AS 16 ENTRADAS DO MC ATIVAM CONFORME SINAIS RECEBIDOS DIRETAMENTE NA ENTRADA DO MCP23017	66
FIGURA 4.10 – TESTE FINAL DO FUNCIONAMENTO DO AUTOSIMOD SOB AS LÓGICAS DE ATUAÇÃO DO RTAC	67
FIGURA 4.11 – ANTIGO MÓDULO SIMULADOR ANALÓGICO UTILIZADO NA COPEL	68
FIGURA 4.12 – NOVO MÓDULO SIMULADOR DIGITAL DO SISTEMA DE SERVIÇOS AUXILIARES DE UMA SUBESTAÇÃO UTILIZADO NA COPEL	69

LISTA DE TABELAS

1	FONTES DE ENERGIA DOS SERVIÇOS AUXILIARES DA CTN	21
2	PROTOCOLOS DE CONVERSAÇÃO SUPOSTADOS PELO RTAC . . .	25

LISTA DE ABREVIATURAS E SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
CA	Corrente Alternada
CC	Corrente Contínua
CI	Circuito Integrado
COPEL	Companhia Paranaense de Energia Elétrica
CTN	Curitiba Norte
ONS	Operador Nacional do Sistema Elétrico
PCB	Placa de Circuito Impresso
RPi	Raspberry Pi
RTAC	Controlador de Automação em Tempo Real
SE	Subestação
SA	Serviços Auxiliares
SACA	Serviços Auxiliares de Corrente Alternada
SACC	Serviços Auxiliares de Corrente Contínua
VAC	Volts em Corrente Alternada
VCC	Volts em Corrente Contínua

SUMÁRIO

1	INTRODUÇÃO	15
1.1	JUSTIFICATIVA	16
1.2	OBJETIVOS	17
1.2.1	Objetivo Geral	17
1.2.2	Objetivos Específicos	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	INTRODUÇÃO À SUBESTAÇÕES	18
2.2	SUBESTAÇÃO ABAIXADORA CURITIBA NORTE (CTN)	19
2.2.1	Equipamentos da subestação	19
2.2.2	Serviços auxiliares da subestação	20
2.2.3	<i>Real-time Automation Controller</i> (RTAC)	23
2.2.4	Automatismo da subestação	25
2.3	PROTOCOLOS	28
2.3.1	Protocolo de comunicação I2C	28
2.4	COMPUTADOR EMBARCADO	29
2.4.1	Raspberry PI	29
2.4.1.1	Raspberry PI B+	30
2.4.2	Raspbian	31
2.4.3	Linguagem de programação Python	31
3	METODOLOGIA	33
3.1	DESENVOLVIMENTO DO HARDWARE	35
3.1.1	Interface de estados (ME - Módulo de Estados)	35
3.1.2	Interface de comandos (MC - Módulo de Comandos)	38
3.2	COMUNICAÇÃO VIA PROTOCOLO I2C	41
3.3	DESENVOLVIMENTO DO SOFTWARE EMBARCADO	43
3.3.1	Tela de início e carregamento de configuração	44
3.3.2	Tela de cadastro de equipamentos	44
3.3.3	Tela de mapeamento de portas	46

	14
3.3.4 Tela principal de simulação	48
3.3.5 Tela de criação de scripts de simulação	50
3.3.6 Telas de exibição gráfica da SE	53
3.3.7 Interação hardware-software	54
4 TESTES E RESULTADOS	59
4.1 TESTES DE HARDWARE	59
4.2 TESTES DE SOFTWARE	60
4.3 INTEGRAÇÃO HARDWARE-SOFTWARE	61
4.3.1 Teste de funcionamento das saídas do ME	62
4.3.2 Teste de funcionamento das entradas do MC	63
4.4 TESTE E RESULTADOS FINAIS	66
4.5 COMPARATIVO ENTRE SIMULADOR ANALÓGICO E O PROTÓTIPO DESENVOLVIDO	67
5 CONCLUSÃO	70
5.1 TRABALHOS FUTUROS	72
5.1.1 Transferência de Conhecimento e Protocolo de Implementação	72
REFERÊNCIAS	74

1 INTRODUÇÃO

A energia elétrica pode ser descrita como um dos insumos básicos do desenvolvimento da civilização, uma vez que esta influencia diretamente em todos os avanços tecnológicos, sociais e econômicos do planeta. A manutenção e o abastecimento dos processos industriais e produtivos da sociedade são proporcionados pela eletricidade, havendo portanto uma grande responsabilidade na entrega contínua desse insumo à população. Para que isso ocorra, de uma maneira geral, são necessárias etapas que vão desde a geração de energia através de fontes renováveis ou não, e a transmissão em alta tensão para polos regionais, até a distribuição de energia realizada pelas subestações de concessionárias locais.

Atualmente nas subestações, as concessionárias utilizam-se da automação dos processos de comando, estado, medidas e funções nos sistemas de proteção, controle supervisão, aquisição de dados e serviços auxiliares. Essa automação tem por objetivo diminuir tanto gastos com recursos operacionais de manutenção quanto permitir um controle eficiente do sistema elétrico da subestação, seja este feito através de um centro de controle remoto ou por acionamentos autônomas na planta. No caso dos serviços auxiliares, é válido destacar que possui um papel vital no funcionamento da subestação, sendo um sistema que compõe dispositivos como cargas, fontes de alimentação e subsistemas de interligação em corrente contínua e corrente alternada que são imprescindíveis para operação dos principais equipamentos de uma subestação. Devido a isto, existe a necessidade de garantir a confiabilidade de funcionamento deste sistema, sendo utilizadas lógicas autônomas que condicionam o acionamento de disjuntores que por sua vez acionam relés de potência que permitem a condução de energia pelas fontes e conseqüente energização das barras com cargas essenciais e não essenciais da subestação.

Na subestação CTN, o equipamento responsável pela automação é o RTAC. Todas as leituras de estados e alarmes de equipamentos da subestação são concentrados e enviados a ele, que tomará decisões de acionar ou não equipamentos presentes na planta, seguindo uma lógica de automação previamente programada. Tal lógica de acionamentos deve ser anteriormente testada e conferida para que a confiabilidade na automação da subestação seja garantida. Para tanto, uma simulação dos cenários possíveis de operação do RTAC deve ser realizada logo no comissionamento da SE,

efetuando a conferência de todos os acionamentos autônomos que este realiza nas diversas circunstâncias de alarmes e estados simulados.

Desta forma, este trabalho visa desenvolver um protótipo de simulação que consiga reproduzir e enviar de forma controlada os sinais de alarme e estado simulados de fontes e cargas elétricas de uma subestação ao RTAC, para que então sejam lidas suas saídas de acionamento e a conferência da lógica de automação da subestação seja realizada. Para tanto, será desenvolvida uma placa modular com expansores de portas (MCP23017) obtendo-se até 32 entradas e saídas digitais, as quais serão utilizadas para enviar alarmes e estados simulados (saídas) e receber os sinais de acionamentos autônomos do RTAC (entradas). O controle das saídas e leitura das entradas será efetuado através de um computador embarcado (Raspberry Pi), que estará conectado diretamente ao módulo expensor de portas e possibilitará o registro e administração gráfica das possibilidades de comutação e estado das portas de simulação, viabilizando o teste dos diversos cenários de acionamentos implementados no automatismo dos serviços auxiliares, que garantem o perfeito funcionamento da subestação.

1.1 JUSTIFICATIVA

A necessidade de manter um alto nível de confiabilidade na comunicação e interpretação de dados que atuam diretamente nos processos e acionamentos dos equipamentos em uma subestação de energia elétrica, tem se tornado um item cada vez exigido por órgãos nacionais regulamentadores no Brasil, como a ANEEL e a ONS. Entende-se que as concessionárias de energia que administram as subestações, devem possuir métodos eficazes e redundantes de operação das mesmas, de forma a não interferir, ou interferir o mínimo possível, no bom funcionamento dos circuitos de transmissão e distribuição de energia elétrica do Sistema Interligado Nacional.

Para tanto, a área de automação, proteção e controle instrumental da COPEL Geração e Transmissão S.A, verificou a carência na análise da lógica de atuação autônoma do sistema de serviços auxiliares da subestação CTN. Tal lógica é implementada por um equipamento conhecido por RTAC (Controlador de Automação em Tempo Real - SEL 2243), que tem por objetivo, analisar estados e alarmes da planta e acionar equipamentos atuadores, que neste caso, compõe o sistema de serviços auxiliares da

SE.

O intuito é desenvolver um protótipo que seja capaz de testar a lógica de automatismo embarcada do RTAC, verificando como este equipamento atuará em variados cenários simulados, e nos casos de ocorrência de falha na lógica, o simulador será capaz de identificar e exibir graficamente o erro ocorrido, possibilitando a previsão planos de ação e correção do sistema sem prejudicar o bom funcionamento da subestação e da operação das linhas de transmissão do Sistema Nacional Interligado.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Desenvolver e implementar um simulador digital que reproduza e verifique os diferentes cenários lógicos e a atuação do controlador de automação em tempo real (RTAC) da subestação Curitiba Norte. O simulador deverá ser utilizado para a realização de testes da operacionalidade do RTAC, visando a identificação e correção de possíveis falhas na lógica de automatismo pertinente aos serviços auxiliares da subestação.

1.2.2 Objetivos Específicos

Para facilitar o entendimento do desenvolvimento prático deste projeto, se fez necessária uma abordagem do conteúdo através de objetivos específicos, sendo estes citados a seguir:

- levantar e estudar os dados técnicos a respeito da subestação CTN e de seu sistema de serviços auxiliares e automatismo de acionamentos;
- levantar as necessidades e melhorias a serem realizadas na atual ferramenta de testes e ensaios do automatismo de SA da COPEL;
- implementar o protocolo I2C na comunicação entre RPi e o módulo expensor de portas através de programação em Python;
- proporcionar a possibilidade de configuração flexível e paralela das portas de simulação;
- comparar e apontar funcionalidades e melhorias do novo protótipo de simulação com o antigo utilizado pela COPEL.

2 FUNDAMENTAÇÃO TEÓRICA

Dentro das subestações, em meio aos diversos subsistemas existentes, destacam-se os de serviços auxiliares, que compreendem todos os equipamentos como disjuntores, chaves seccionadoras, relés de potência, dentre outros que realizam atuações e manobras dentro da planta. Devido à enorme demanda por energia elétrica existente hoje na sociedade, o emprego de circuitos redundantes e automatismo nas atuações dos SA se faz indispensável, por ser esse um sistema essencial para o funcionamento da SE. Para que se possa colocar esse sistema em funcionamento autônomo, primeiramente, deve-se garantir uma confiabilidade muito alta na lógica de automatismo utilizada. Para tanto, realizar simulações dessa lógica no comissionamento de novas SEs é algo que vem sendo feito pelas concessionárias de energia.

A COPEL utiliza atualmente um sistema de simulação completamente analógico, em que se faz uso de botoeiras mecânicas, relés e LEDs para simular, enviar, e receber respectivamente sinais de alarme, estado e atuação do controlador de automação RTAC. A computadorização dessa atual ferramenta utilizada pela COPEL é um aperfeiçoamento necessário, visto que é um sistema desprovido de memória e recursos digitais para a realização das simulações. Para tanto, o emprego de um computador embarcado Raspberry Pi junto à uma placa expansora de portas se comunicando em protocolo I2C, aliado ainda à um *software* com interface gráfica para controle, traria novas funcionalidades e as melhorias necessárias para esse sistema.

2.1 INTRODUÇÃO À SUBESTAÇÕES

As subestações são soluções que compõem o sistema de proteção, controle, transmissão e distribuição de energia de alta potência de uma fonte geradora à consumidora. Uma subestação tem a função de transformar a tensão, agindo como pontos de entrega para a demanda (AGERADORA, 2016). As unidades geradoras de energia elétrica normalmente estão situadas distantes dos centros consumidores, ou seja, em pequenas cidades e zonas industriais. Desta forma, toda a energia produzida deve viajar longas distâncias através de um sistema de transmissão. Mas antes de chegar aos centros de consumo, essa a energia passa por uma subestação abaixadora, que tem a função de abaixar o nível de tensão, para que esta esteja em um nível adequado para o abastecimento de pequenas cidades ou consumidores de grande porte (SATO,

2013).

Existem diversas classificações para as subestações, como por exemplo quanto ao tipo de instalação, quanto à sua função no sistema, quanto ao nível de tensão, dentre outras. Para atender ao objetivo e escopo deste projeto, este trabalho se abstém de citar as diversas características e informações a respeito dos vários tipos de subestações, voltando o seu foco à subestação Curitiba Norte que se enquadra no tipo de subestação abaixadora.

2.2 SUBESTAÇÃO ABAIXADORA CURITIBA NORTE (CTN)

Localizada na periferia de Curitiba, mais precisamente no subúrbio do município de Almirante Tamandaré, a subestação CTN realiza a função de abaixar a tensão da rede de transmissão da Companhia Paranaense de Energia (COPEL) de 230 kV para 138 kV, para que essa energia chegue em níveis de consumo adequados para as proximidades dos centros urbanos de Curitiba e região. Existem diversos motivos e vantagens para que seja realizada essa função de abaixamento de tensão, dentre eles, evitar inconvenientes para a população como rádio interferência, campos magnéticos intensos, dentre outros (MUZY, 2012). Outro fator muito importante a ser levado em consideração é o risco de acidentes graves envolvendo as linhas de alta tensão que transitam em meio aos centros urbanos.

Inaugurada em Junho de 2016, a CTN (ver Figura 2.1) possui equipamentos modernos tanto em seus sistemas de potência quanto em seus sistemas de controle. Tais ambientes, equipamentos e sistemas são padronizados segundo a Agência Nacional de Energia Elétrica (ANEEL) e também todas as operações de manutenção realizadas na planta seguem padronização aplicada segundo o Operador Nacional do Sistema Elétrico (ONS).

2.2.1 Equipamentos da subestação

Uma subestação é composta por diversos equipamentos, dentre eles se destacam os disjuntores, chaves seccionadoras, transformadores, relés, controladores lógicos programáveis, para-raios e resistores de aterramento (SUCKOW, 1999). Incluindo todos estes citados, a CTN também possui reguladores de tensão, medidores de potência/-tensão/corrente dentre outros. Na Figura 2.1 é possível visualizar os dois principais

FIGURA 2.1 – SUBESTAÇÃO CURITIBA NORTE - CTN



Fonte: Os Autores (2018)

equipamentos da CTN, os transformadores de tensão de 230 kV para 138 kV. A grande maioria destes equipamentos possuem, de modo geral, sensores analógicos e/ou digitais que, dependendo de sua aplicação, necessitam ser monitorados em tempo real para que a operação da subestação aconteça de forma controlada e segura. Todos os alarmes e medidas analógicas são apresentados nas instalações onde se encontram os equipamentos e enviados a um equipamento concentrador, que será abordado em detalhes futuramente neste trabalho.

2.2.2 Serviços auxiliares da subestação

Os serviços auxiliares (SA) são sistemas compostos por todos os dispositivos e cargas necessárias para o funcionamento dos equipamentos principais de uma subestação. Tal sistema abrange fontes de alimentação e subsistemas de manobra compostos por transformadores, painéis, retificadores, relés, dentre outros equipamentos essenciais para o funcionamento da SE. Basicamente, os serviços auxiliares podem ser subdivididos em dois subsistemas, um alimentado por corrente alternada e outro por corrente contínua. Todos os equipamentos que compõe os serviços auxiliares necessitam de uma alimentação de baixa tensão para funcionar, 110 ou 220 V para o subsistema de corrente alternada e 132 V para o subsistema de corrente contínua.

Na CTN, os serviços auxiliares em corrente alternada é alimentado por uma tensão trifásica de 220 V, possuindo três possíveis fontes de energia para alimentar as cargas essenciais e não essenciais da SE, como pode ser visto na Figura 2.2. Em último caso, existe ainda um banco de baterias para o advento de uma falha geral na alimentação dos serviços auxiliares em CC, totalizando quatro possíveis fontes que são mostradas na Tabela 1.

TABELA 1 – FONTES DE ENERGIA DOS SERVIÇOS AUXILIARES DA CTN

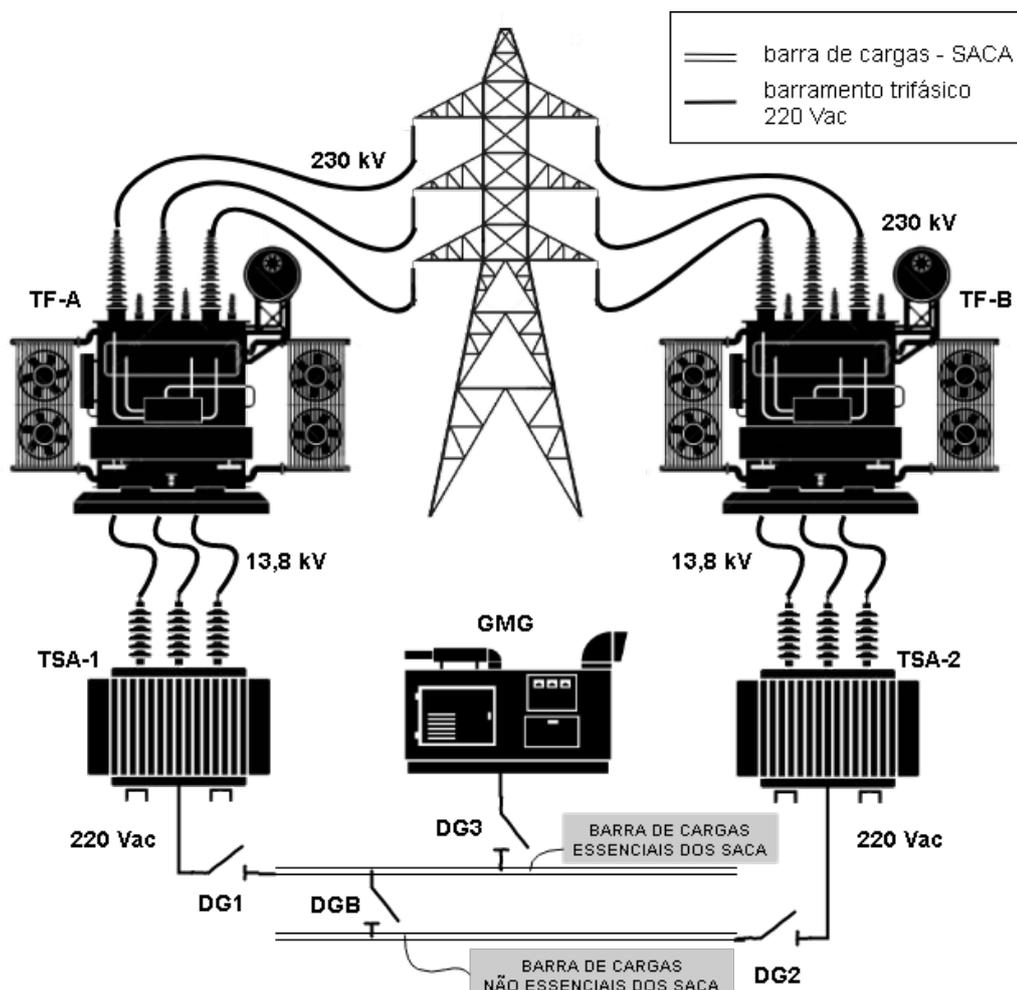
Sigla	Nome
TSA-1	Transformador de potência 1
TSA-2	Transformador de potência 2
GMG	Gerador de emergência
BNC	Banco de baterias

Fonte: Os Autores (2018)

Assegurar que a alimentação dos serviços auxiliares continue mesmo após uma queda completa de energia na planta, é uma das necessidades primárias de uma subestação. Qualquer abstinência da tensão de alimentação dos serviços auxiliares, mesmo que temporária, causaria um “apagão” na planta, pois os transformadores abririam de imediato. Conseqüentemente, os SA estão sempre conectados a um banco de baterias que garantem o seu funcionamento ininterrupto (CALONE, 2009). A Figura 2.2 exibe as fontes conectadas a partir de duas linhas de transmissão de 230 kV que alimentam a CTN. Essas linhas abastecem os terminais primários de transformadores de potência 230/138 kV, que são responsáveis pelo abaixamento de tensão nesse sistema.

Uma vez com a tensão em 138 kV, o circuito se conecta à um segundo transformador que tem por objetivo reduzir mais uma vez a tensão para o patamar de 220 V, nível este que energizará as barras essenciais. Essas barras interconectam todo o sistema de serviços auxiliares da subestação, como mostram as Figuras 2.3 e 2.4, evidenciando as barras de força que se conectam à todas as cargas essenciais e não essenciais do SACA e cargas do SACC. A alimentação das cargas em corrente contínua depende diretamente da disponibilidade das fontes em CA, visto que a alimentação de todo o serviço auxiliar em CC provém da retificação das fontes em CA. Tanto os serviços auxiliares em CA quanto os serviços auxiliares em CC possuem alimentação redundante conforme é possível observar nas Figuras 2.3 e 2.4.

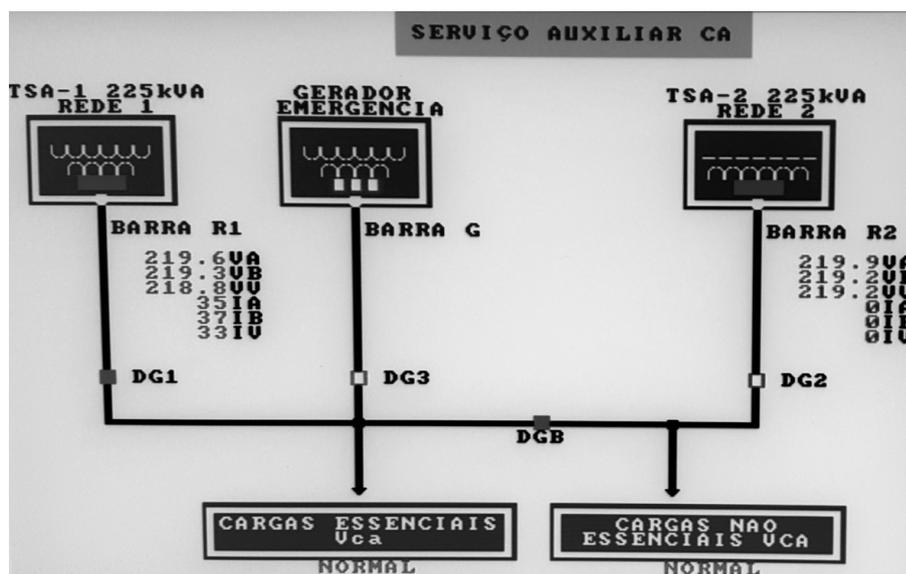
FIGURA 2.2 – DIAGRAMA DE ALIMENTAÇÃO DOS SA - SACA - CTN



Fonte: Os Autores (2018)

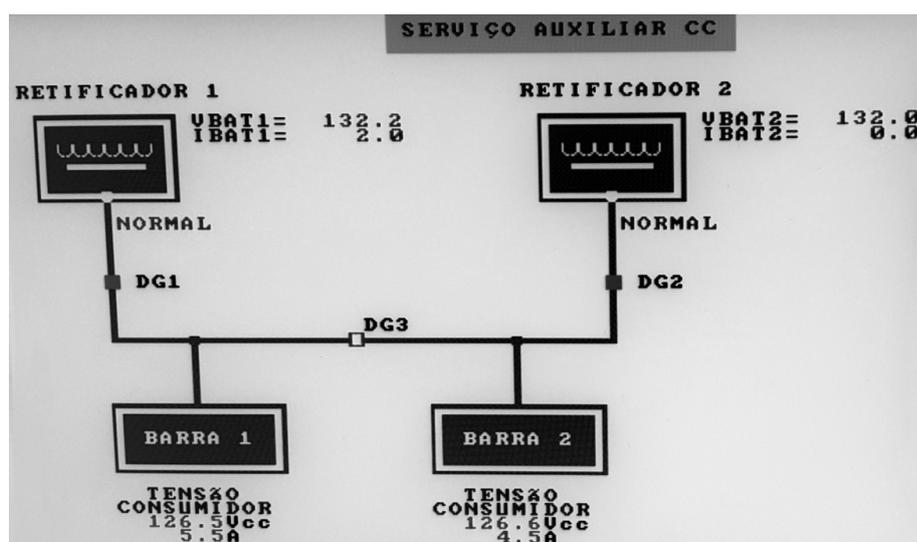
A alimentação dos serviços auxiliares em CC apresenta ainda redundância de retificadores (ver Figura 2.5), para o caso de um destes equipamentos vir a falhar, o outro atuará em seu lugar. Se em determinado momento ocorrer um cenário de falha na subestação, em que esta acarretou uma falta de energia nas principais fontes de CA (TSA-1 e TSA-2), ou nos retificadores em CC, uma atuação rápida por parte do controle autônomo da subestação deve ser desempenhada. Neste caso, uma lógica de acionamento envolvendo os disjuntores DG1, DG2, DG3 e DGB seria executada, substituindo a fonte ou retificador que falhou por outro(a), visando manter os serviços auxiliares em funcionamento ininterrupto.

FIGURA 2.3 – DIAGRAMA UNIFILAR DE ALIMENTAÇÃO - SACA - CTN



Fonte: Os Autores (2018)

FIGURA 2.4 – DIAGRAMA UNIFILAR DE ALIMENTAÇÃO - SACC - CTN

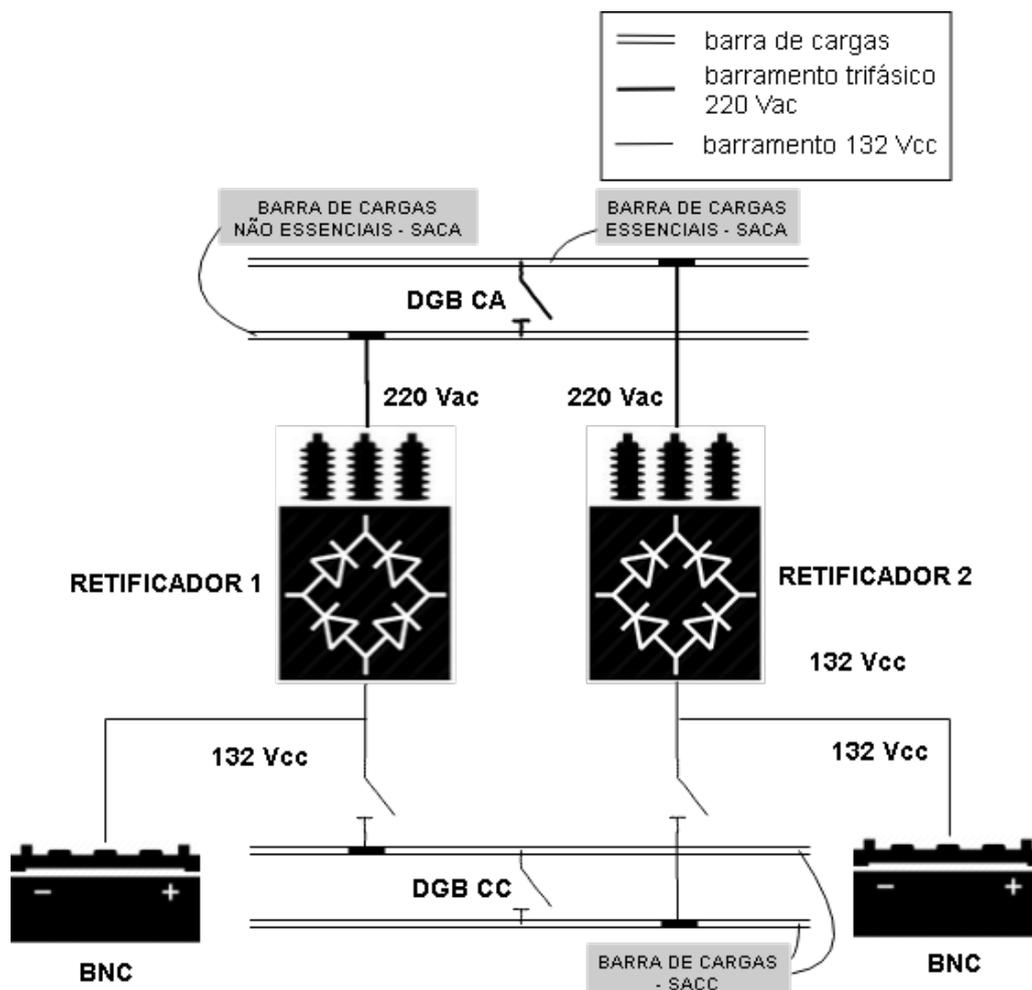


Fonte: Os Autores (2018)

2.2.3 Real-time Automation Controller (RTAC)

Na subestação CTN, o equipamento que realiza todo o automatismo dos serviços auxiliares é o RTAC. Trata-se de um equipamento adequado para ser utilizado em subestações e sistemas industriais de controle e automação. O RTAC está conectado a todos os equipamentos que possuem algum módulo de leitura que forneça medidas digitais ou analógicas ou entradas de atuação na SE. Ele fornece controle completo da subestação, possuindo compatibilidade com diversos protocolos de comunicação e flexibilidade em questão de programação de sua lógica de atuação, sendo capaz

FIGURA 2.5 – DIAGRAMA DE ALIMENTAÇÃO - SACC - CTN



Fonte: Os Autores (2018)

de ler e atuar nos mais diversos tipos de dispositivos. Os protocolos de conversação suportados na comunicação com o RTAC são exibidos na Tabela 2. O RTAC modelo SEL-2243 que está dentro da casa de comandos da CTN, é exibido na Figura 2.6. As entradas de estados da SE preenchem dezenas de portas dos sete últimos módulos de entrada presentes no equipamento, assim como as saídas de atuação no primeiro módulo de portas à esquerda da imagem. Como é possível notar nesta imagem, existem muito mais entradas de estados que saídas de atuação na CTN. Devido a isso, o simulador deverá comportar muito mais saídas do que entradas, pelo entendimento de que todas as saídas do simulador serão conectadas às entradas do RTAC, assim como as entradas, que serão conectadas às saídas de atuação do RTAC. Os estados e alarmes pertinentes ao SACA e SACC que serão simulados e enviados ao RTAC, compõe apenas sinais no escopo digital. Portanto, não existe necessidade de conversação entre o simulador e o RTAC em nenhum dos protocolos citados na Tabela 2.

TABELA 2 – PROTOCOLOS DE CONVERSAÇÃO SUPORTADOS PELO RTAC

Protocolos (lado cliente)	Protocolos (lado servidor)
SEL Fast Messaging	SEL Fast Messaging
SEL ASCII	DNP3 serial
DNP3 serial	DNP3 LAN/WAN
DNP3 LAN/WAN	Modbus RTU
Modbus® RTU	Modbus TCP
Modbus TCP	EC 60870-5-101/104
IEC 60870-5 101/104	LG 8979
LG 8979	SES-92
CP 2179	Syslog
IEEE C37.118.1a-2014	IEEE C37.118.1a-201
synchrophasors	synchrophasors
IEC 61850 MMS and MMS	IEC 61850 MMS and MMS
Client File Services	Server File Service
FTP/SFTP	FTP/SFTP

Fonte: (SCHWEITZER, 2018)

2.2.4 Automatismo da subestação

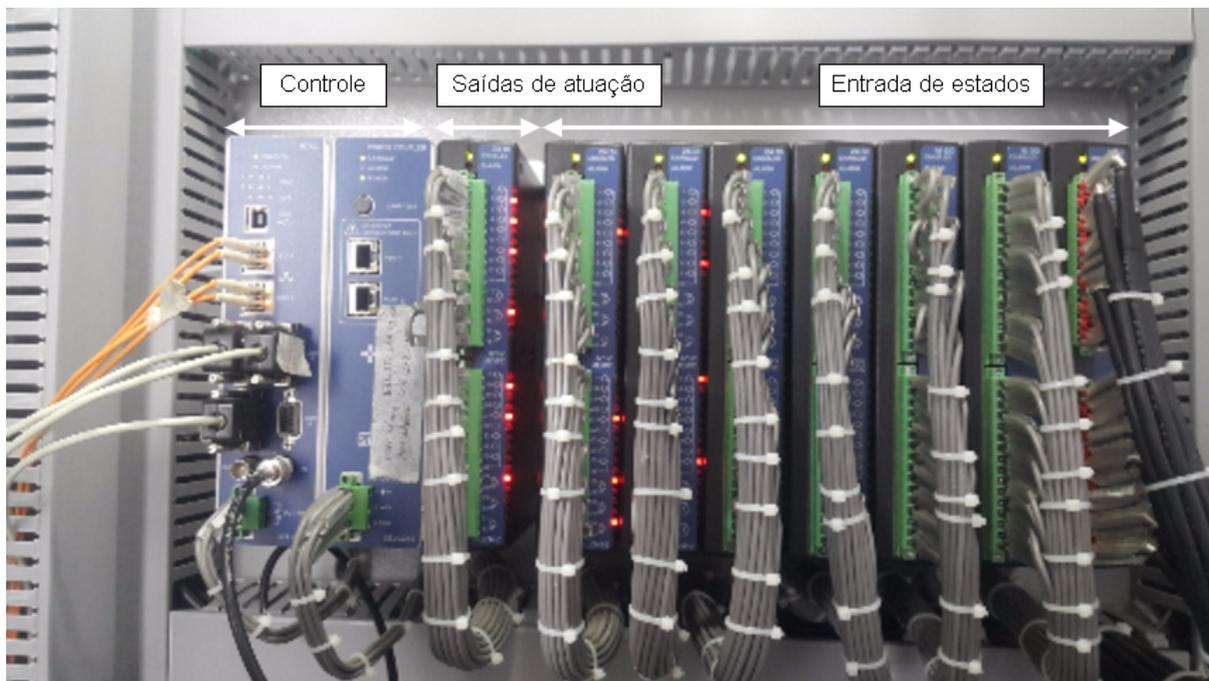
A operação de uma subestação é bastante complexa, devido ao elevado número de variáveis (estados e alarmes) a serem manipulados. Em vista disso, a automação de subestações é imprescindível para a qualidade no fornecimento de energia elétrica, no sentido em que reduz a quantidade e tempo de interrupções realizadas na planta através da atuação de um sistema de controle automatizado em tempo real. Todas as leituras tanto de sensores digitais quanto analógicos, estados e alarmes da planta da CTN, são direcionados ao equipamento concentrador RTAC, que analisa essas informações de entrada em tempo real. Este equipamento, de forma automatizada, irá atuar ou não na planta dependendo de uma lógica previamente configurada. Por “atuação” entende-se, abrir ou fechar relés, ativar/desativar equipamentos, conectar/desconectar circuitos.

As lógicas de acionamento serão abordadas posteriormente neste trabalho com maior abundância de detalhes. Na Figura 2.7, é possível visualizar, conforme evidenciado, o estado dos disjuntores da alimentação dos serviços auxiliares em CC (sistema essencial para o funcionamento da SE, abordado no próximo tópico). Esses estados são exibidos em um terminal dentro da casa de controle da SE, sendo leituras provindas diretamente do RTAC.

A utilização de sistemas e fontes redundantes na alimentação dos SA é indis-

pensável, visto que este tem papel fundamental no funcionamento da SE e impacto direto no fornecimento de energia elétrica em toda uma determinada região para o caso de falhar.

FIGURA 2.6 – RTAC SEL-2243 DA SCHWEITZER NA SUBESTAÇÃO CTN



Fonte: Os Autores (2018)

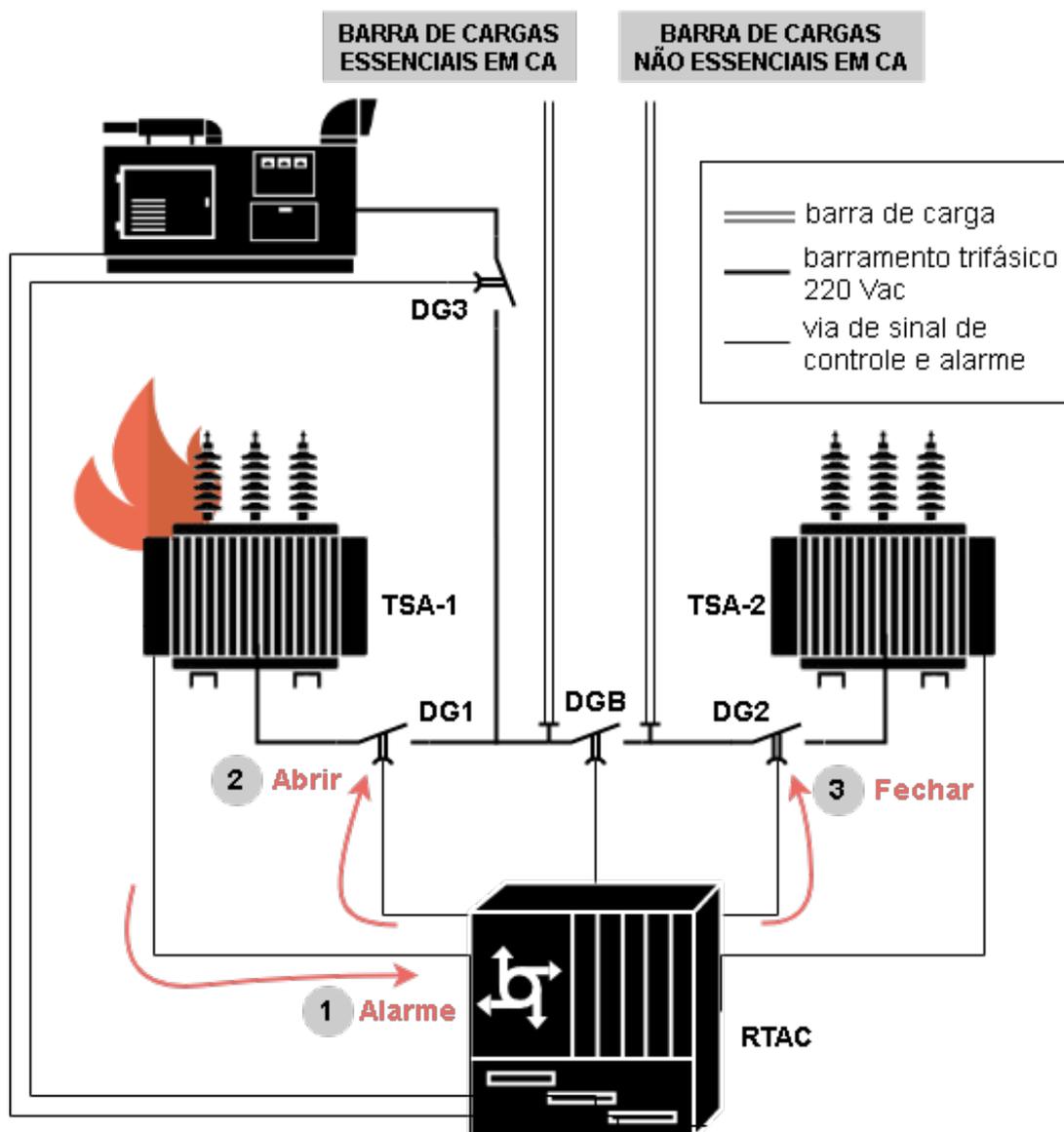
FIGURA 2.7 – ALARMES E ESTADOS DIGITAIS EM TEMPO REAL DOS EQUIPAMENTOS DA SUBESTAÇÃO CTN. DESTAQUE PARA ESTADOS DOS DISJUNTORES DA ALIMENTAÇÃO DOS SERVIÇOS AUXILIARES EM CC

UNID	UNID	ALR	ANA	DIG	FUN	EUE	R	?	↑	↓	
991	CA	DCI		FECHADO		931	CB-1	NEG/TERRA			NORMAL
992	CA	DEF		NORMAL		932	CB-1	SOBRE TENSÃO			NORMAL
993	CA	DEF		ABERTO		933	CB-1	SUB TENSÃO			ALARME
994	CA	DEF		NORMAL		934	CB-1	SCARREGADOR BAT			NORMAL
995	CA	DEF		FECHADO		935	CB-1	ANOMALIA			NORMAL
996	CA	DEF		NORMAL		936	CB-1	POS/TERRA			NORMAL
997	CA	DEF		NORMAL		937	CB-1	NEG/TERRA			NORMAL
998	CA	DEF		NORMAL		938	CB-1	SOBRE TENSÃO			NORMAL
999	CA	DEF		NORMAL		939	CB-1	SUB TENSÃO			NORMAL
1000	CA	DEF		NORMAL		940	CB-1	GPS SINC			NORMAL
1001	CA	DEF		NORMAL		941	CB-1	MULTIIMEDIDOR			NORMAL
1002	CA	DEF		NORMAL		942	CB-1	MULTIIMEDIDOR			NORMAL
1003	CA	DEF		NORMAL		943	CB-1	MULTIIMEDIDOR			NORMAL
1004	CA	DEF		NORMAL		944	CB-1	MULTIIMEDIDOR			NORMAL
1005	CA	DEF		NORMAL		945	CB-1	MULTIIMEDIDOR			NORMAL
1006	CA	DEF		NORMAL		946	CB-1	MULTIIMEDIDOR			NORMAL
1007	CA	DEF		NORMAL		947	CB-1	MULTIIMEDIDOR			NORMAL
1008	CA	DEF		NORMAL		948	CB-1	MULTIIMEDIDOR			NORMAL
1009	CA	DEF		NORMAL		949	CB-1	MULTIIMEDIDOR			NORMAL
1010	CA	DEF		NORMAL		950	CB-1	MULTIIMEDIDOR			NORMAL
1011	CA	DEF		NORMAL		951	CB-1	MULTIIMEDIDOR			NORMAL
1012	CA	DEF		NORMAL		952	CB-1	MULTIIMEDIDOR			NORMAL
1013	CA	DEF		NORMAL		953	CB-1	MULTIIMEDIDOR			NORMAL
1014	CA	DEF		NORMAL		954	CB-1	MULTIIMEDIDOR			NORMAL
1015	CA	DEF		NORMAL		955	CB-1	MULTIIMEDIDOR			NORMAL
1016	CA	DEF		NORMAL		956	CB-1	MULTIIMEDIDOR			NORMAL
1017	CA	DEF		NORMAL		957	CB-1	MULTIIMEDIDOR			NORMAL
1018	CA	DEF		NORMAL		958	CB-1	MULTIIMEDIDOR			NORMAL
1019	CA	DEF		NORMAL		959	CB-1	MULTIIMEDIDOR			NORMAL
1020	CA	DEF		NORMAL		960	CB-1	MULTIIMEDIDOR			NORMAL
1021	CA	DEF		NORMAL		961	CB-1	MULTIIMEDIDOR			NORMAL
1022	CA	DEF		NORMAL		962	CB-1	MULTIIMEDIDOR			NORMAL
1023	CA	DEF		NORMAL		963	CB-1	MULTIIMEDIDOR			NORMAL
1024	CA	DEF		NORMAL		964	CB-1	MULTIIMEDIDOR			NORMAL
1025	CA	DEF		NORMAL		965	CB-1	MULTIIMEDIDOR			NORMAL
1026	CA	DEF		NORMAL		966	CB-1	MULTIIMEDIDOR			NORMAL
1027	CA	DEF		NORMAL		967	CB-1	MULTIIMEDIDOR			NORMAL
1028	CA	DEF		NORMAL		968	CB-1	MULTIIMEDIDOR			NORMAL
1029	CA	DEF		NORMAL		969	CB-1	MULTIIMEDIDOR			NORMAL
1030	CA	DEF		NORMAL		970	CB-1	MULTIIMEDIDOR			NORMAL

Fonte: Os Autores (2018)

A existência de falhas que venham a incorrer no bloqueio total de funcionamento da CTN por exemplo, acarretariam em prejuízo imensurável à concessionária de energia elétrica detentora de sua concessão (COPEL) e também prejuízos na qualidade da energia entregue a população regional abastecida por ela.

FIGURA 2.8 – CENÁRIO HIPOTÉTICO DE FALHA NO TRANSFORMADOR PRINCIPAL DE SERVIÇOS AUXILIARES E SUBSEQUENTES ATUAÇÕES DO RTAC



Fonte: Os Autores (2018)

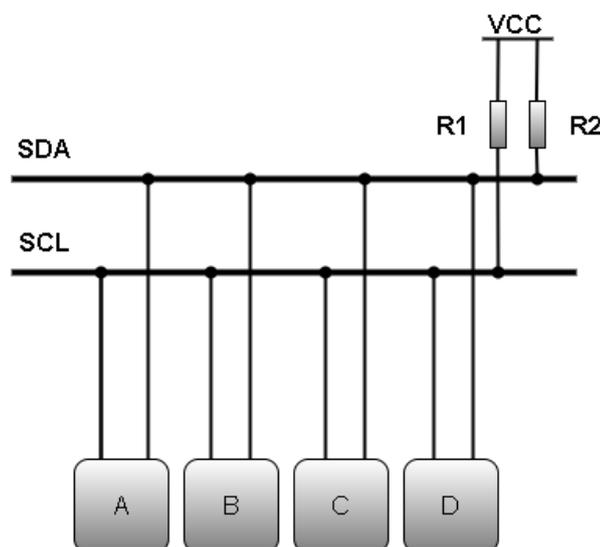
A Figura 2.8 mostra um processo de ativação a ser executado por uma das lógicas autônomas embarcadas no RTAC, que utiliza a redundância aplicada às barras da subestação e garante o funcionamento do sistema sem prejuízo na entrega final de energia.

2.3 PROTOCOLOS

2.3.1 Protocolo de comunicação I2C

O protocolo I2C é utilizado na comunicação de um ou mais dispositivos utilizando apenas dois fios. Para isso, cada periférico deve ter um ID (número de identificação) predefinido, ou o endereço de um dispositivo exclusivo, dessa forma o mestre da comunicação pode escolher com quem estará se comunicando. Os dois fios são chamados de *Serial Clock* (SCL) e *Serial Data* (SDA). A linha SCL é o sinal de relógio, responsável por sincronizar a transferência de dados entre os dispositivos conectados ao barramento I2C, sendo este gerado pelo mestre. A outra linha, SDA é responsável pelo transporte de dados (CARVALHO, 2016). A Figura 2.9 exibe um diagrama de conexões básico de um esquema de conversação I2C, onde os dispositivos A, B, C e D se comunicam entre si através do barramento SDA, estando sincronizados pelo barramento de relógio SCL.

FIGURA 2.9 – ESQUEMA BÁSICO DE COMUNICAÇÃO EM PROTOCOLO I2C

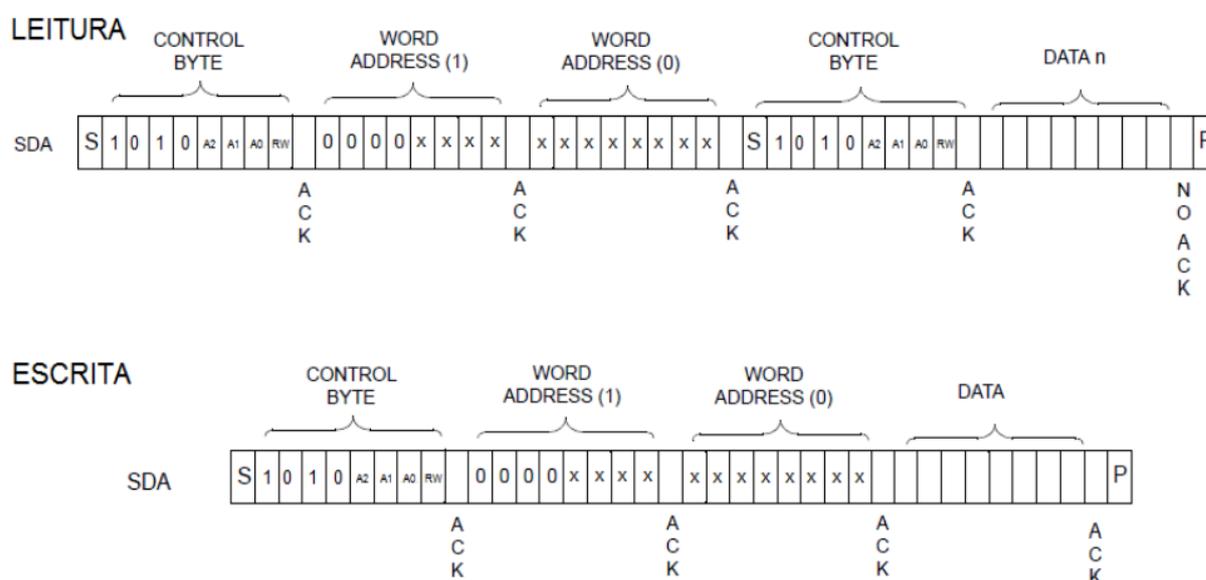


Fonte: Os Autores (2018)

A Figura 2.10 apresenta a estrutura de dados do protocolo o I2C. A transmissão é serial e bidirecional, podendo ser feita em três taxas de comunicação. A primeira delas é a transmissão de 100 kbps no modo *Standard*, a segunda é de 400 kbps no modo *Fast* e por fim de até 3,4 Mbps no modo *High Speed*. Os dados são transferidos entre o dispositivo principal e os dispositivos "escravos" em uma sequência de 8 bits. Dessa forma, após a chegada de uma condição de início, o cabeçalho de endereçamento

apresenta nos casos de leitura do protocolo, uma dupla sequência de 8 bits que indica o endereço do dispositivo "escravo" para qual os dados estão sendo enviados. Após esse endereçamento inclui-se 1 bit, chamado de *Acknowledge* e outros 8 bits que indicam, na maioria dos casos, uma sequência de endereçamento interna do dispositivo "escravo". Uma vez dadas essas informações, encaminham-se os bytes que representam os dados a serem transferidos, até a chegada da condição de parada (*Stop bit*) Para o protocolo de escrita apenas é desconsiderado o último pacote de 8 bits e o 1 bit de acknowledge (CARVALHO, 2016).

FIGURA 2.10 – BLOCOS DE 8 BITS DE ESCRITA E LEITURA DO PROTOCOLO I2C



Fonte: (CARVALHO, 2016)

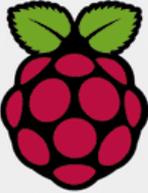
2.4 COMPUTADOR EMBARCADO

2.4.1 Raspberry Pi

O Raspberry Pi foi desenvolvido em fevereiro de 2012 pela fundação Raspberry Pi, originalmente para fins educacionais, para promover o ensino básico da ciência da computação nas escolas e nas faculdades em torno do Reino Unido. Uma de suas características que chama atenção, além da capacidade computacional e do baixo custo, é o seu tamanho, que é aproximadamente o de um cartão de crédito (PATCHAVA; KANDALA; BABU, 2015). As placas RPi usam ARM, tecnologia que é utilizada na placa para reduzir o calor e consumo de energia. Trata-se de um processador multi-núcleo, implementado com SoC e com requisito de alimentação 5 V e 700 mA. (PATCHAVA;

KANDALA; BABU, 2015). A Figura 3.11 mostra uma tabela comparativa entre os modelos de computador Raspberry, disponíveis no mercado.

FIGURA 2.11 – COMPARATIVO ENTRE AS VERSÕES DO RASPBERRY PI MAIS ATUAIS

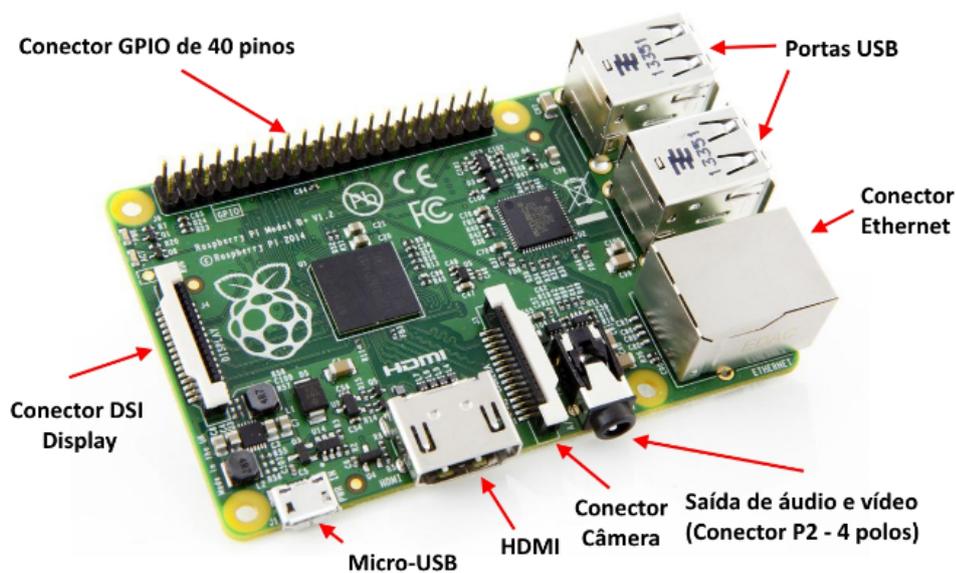
	Raspberry Pi 3 Model B	Raspberry Pi Zero	Raspberry Pi 2 Model B	Raspberry Pi Model B+
Introduction Date	2/29/2016	11/25/2015	2/2/2015	7/14/2014
SoC	BCM2837	BCM2835	BCM2836	BCM2835
CPU	Quad Cortex A53 @ 1.2GHz	ARM11 @ 1GHz	Quad Cortex A7 @ 900MHz	ARM11 @ 700MHz
Instruction set	ARMv8-A	ARMv6	ARMv7-A	ARMv6
GPU	400MHz VideoCore IV	250MHz VideoCore IV	250MHz VideoCore IV	250MHz VideoCore IV
RAM	1GB SDRAM	512 MB SDRAM	1GB SDRAM	512MB SDRAM
Storage	micro-SD	micro-SD	micro-SD	micro-SD
Ethernet	10/100	none	10/100	10/100
Wireless	802.11n / Bluetooth 4.0	none	none	none
Video Output	HDMI / Composite	HDMI / Composite	HDMI / Composite	HDMI / Composite
Audio Output	HDMI / Headphone	HDMI	HDMI / Headphone	HDMI / Headphone
GPIO	40	40	40	40
Price	\$35	\$5	\$35	\$35

Fonte: (SILVA, 2016)

2.4.1.1 Raspberry Pi B+

O Raspberry Pi B+ que está sendo exibido na Figura 2.12, conta com SoC baseado ARM, modelo ARM1176JZF-S, produzido pela Broadcom BCM2835 com 512 MBytes (compartilhada com a GPU) de memória RAM e opera na frequência de 700 MHz. Também conta com uma unidade de processamento gráfico, GPU modelo Video Core IV de 250 MHz e um *slot* para cartão SD, que é usado para inicializar os sistemas operacionais Linux como Raspbian, Pidora e Raspbm. A versão Raspberry Pi B+ tem quatro portas USB 2.0 para conectar os periféricos como, mouse, teclado e adaptador Wi-Fi, uma porta *Ethernet* 10/100 para se conectar a rede e quarenta pinos de entrada e saídas, que podem ser usados para controlar LEDs, *switches*, sensores e outros dispositivos. Uma saída HDMI para se conectar com monitores LCD, projetores e televisores e alguns recursos adicionais, como conector da câmera e saída de áudio, jack de 3,5 mm, estão disponíveis (PATCHAVA; KANDALA; BABU, 2015) para criar a interface de uso e gráfica com o usuário. Esse dispositivo será utilizado no projeto como um controlador dos módulos de comando e estado.

FIGURA 2.12 – MODELO RASPBERRY PI B+



Fonte: (ARDUINOECIA, 2014)

2.4.2 Raspbian

O Raspbian é um sistema operacional gratuito, baseado no Debian, otimizado para o *hardware* Raspberry Pi. Desde 2015 foi oficialmente fornecido pela Fundação Raspberry Pi como o principal sistema operacional para a família Raspberry Pi de computadores de placa única. Raspbian foi criado por Mike Thompson e Peter Green como um projeto independente. A compilação inicial foi concluída em junho de 2012. O sistema operacional ainda está em desenvolvimento ativo, e é altamente otimizado para as CPUs ARM de baixo desempenho (FOUNDATION, 2017).

O modelo RPi que será utilizado no projeto é o B+, plataforma essa que junto ao sistema operacional Raspbian, faz-se possível o desenvolvimento de um ambiente computacional adequado para as necessidades desse projeto, em que será concebido o ambiente de interação entre o simulador digital e o utilizador.

2.4.3 Linguagem de programação Python

Python é uma linguagem de programação interpretada, onde cada linha é compilada e executada, uma por vez. Isso significa que ela não é compilada de forma que todo o texto seja traduzido para linguagem de máquina e posteriormente executado como acontece em outras linguagens. Pode ser descrita como uma linguagem de alto nível, não sendo necessário pagar para sua utilização. As aplicações em Python vão

desde a criação de scripts para *backend* de servidores até aplicações mais sofisticadas, incluindo interface gráfica e interatividade com o usuário. Devido à grande adesão do público desenvolvedor à projetos e aplicações baseadas em Python com plataformas da família Raspberry, criou-se um grande número de bibliotecas escritas e conteúdo extenso disponível na internet a respeito. O desenvolvimento do *software* de simulação dos serviços auxiliares, será todo baseado em Python e utilizará bibliotecas já existentes na comunidade para programação embarcada em Raspberry Pi. Modelagens dos códigos responsáveis pelo funcionamento do protótipo, serão utilizados nos próximos capítulos para facilitar o entendimento do projeto,

3 METODOLOGIA

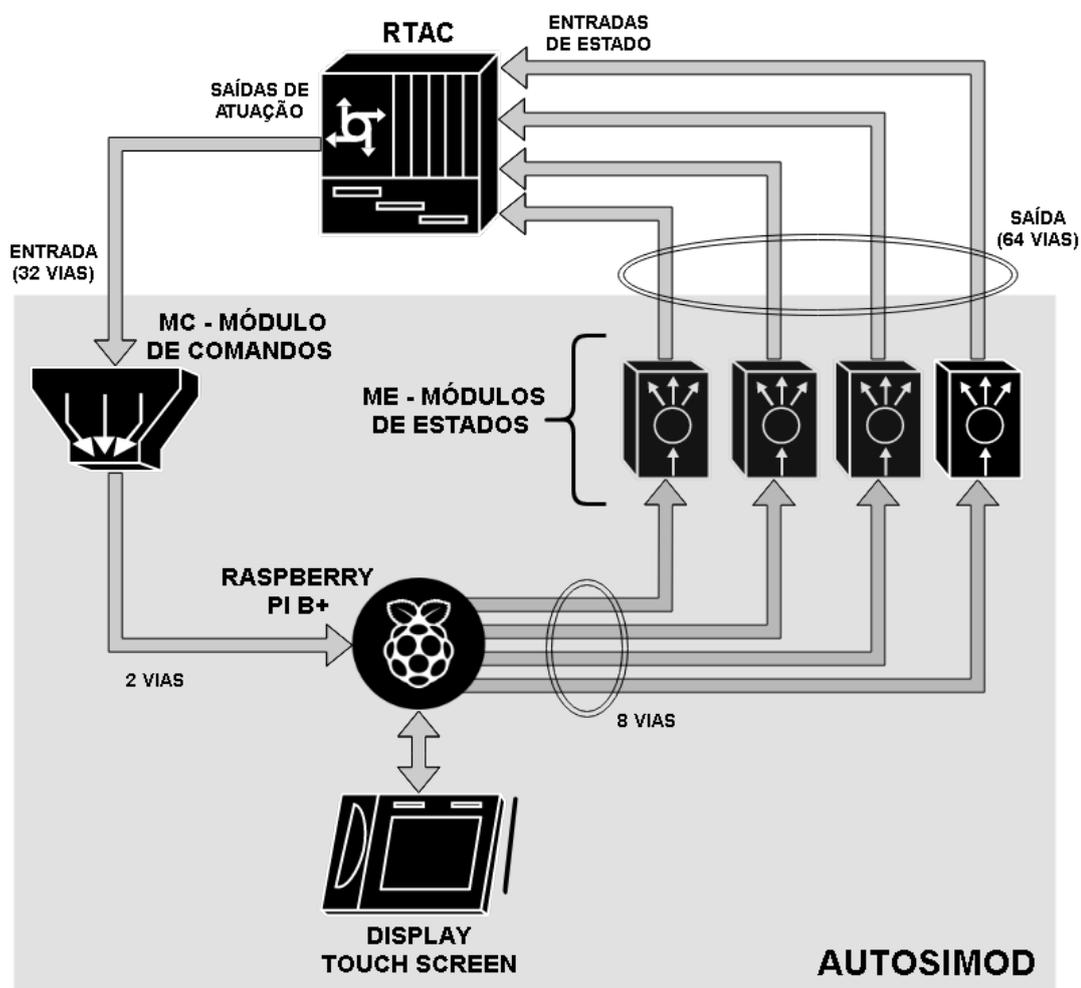
A introdução de tecnologia digital nas subestações, substituindo tecnologias baseadas em modelos analógicos, acrescenta melhorias em todos os sentidos para uma SE, desde melhor compatibilidade com os sistemas modernos até no modo de operação e manutenção de toda ela. Com a automação dos processos de atuação dos serviços auxiliares, o controle de uma subestação é feito com o mínimo de intervenção técnica, resultando em maior eficiência de operação da mesma. O nível de automatismo de uma subestação é diretamente proporcional ao nível de confiabilidade do sistema de controle automatizado. É importante salientar que uma única leitura errônea ou atuação não condizente do sistema, poderia desencadear uma série danos aos equipamentos chegando a acarretar inoperância completa de todas as funções da subestação. Devido a este fato, é evidente a necessidade de testar e avaliar todos os cenários de atuação da lógica de automatismo dos serviços auxiliares de uma subestação.

Uma maneira de avaliar previamente a confiabilidade de uma lógica executada pelo RTAC de uma subestação, seria a simulação digital dos cenários de estados e a consequente leitura e exibição dos acionamentos realizados por essa lógica. Isso mostraria com clareza ao operador da simulação, as possíveis falhas presentes na lógica de automação. Para tanto, se faz necessário um equipamento digital que agregue as características citadas, para que seja implementado já no comissionamento de novas subestações. A Figura 3.1 possui um diagrama elementar do simulador proposto, exibindo as conexões entre este e o RTAC, bem como as ligações internas entre o RPi B+ e os módulos extensores de portas MC e ME. O diagrama apresenta o ciclo de informações transmitidas e interpretadas pela placa modular em conexão ao controlador dos SA, o RTAC. Como pode ser visto, o protótipo funcionará como um replicador e receptor de estados e acionamentos respectivamente emitidos e recebidos pelo controlador, simulando o sistema de SA da subestação.

Toda informação digital transitada entre as entradas e saídas do simulador serão mapeadas via software e exibidas em uma tela *touchscreen* para o utilizador em forma de um *log*. Na interface gráfica com o usuário, será possível criar scripts de simulação que basicamente se traduzem em uma configuração temporizada de acionamentos e desacionamentos das saídas de estados do simulador. O usuário poderá também atuar individualmente em cada saída ao toque de um *click* na tela de

simulação.

FIGURA 3.1 – DIAGRAMA ELEMENTAR DE CONEXÕES INTERNAS DO SIMULADOR E CONEXÕES EXTERNAS COM O RTAC



Fonte: Os Autores (2018)

Efetuada a simulação e envio controlado dos estados e alarmes da SE ao RTAC e logo após, lendo suas portas de atuação, permite identificar as possíveis falhas existentes na lógica embarcada configurada nele. Algumas das simulações que serão realizadas incluem cenários como: simular uma cadeia específica de estados, distorcendo o correto funcionamento dos serviços auxiliares da SE; temporizar aberturas e fechamentos não programados dos disjuntores das barras da subestação; acusar alarmes de subtensão e sobretensão em disjuntores; acusar alarmes de temperatura, simular alarmes e acionamentos em cadeia através de script pré-configurado via interface de software, dentre outros, ficando a cadeia de eventos e acionamentos registrada em um relatório à disposição do usuário via tela em tempo real, ou em arquivo txt.

3.1 DESENVOLVIMENTO DO HARDWARE

3.1.1 Interface de estados (ME - Módulo de Estados)

O módulo de estados tem por objetivo receber os sinais de atuação do RTAC, bem como os sinais de contatos auxiliares de estado de equipamentos de campo. Dentre eles estão os disjuntores, chaves seccionadoras, chaves de painel, relés de proteção ou auxiliares, e outros dispositivos acionadores. Estes sinais que deveriam ser enviados diretamente aos equipamentos, acabam por ser conectados ao ME, para que este desempenhe a leitura dos acionamentos a serem executados pelo RTAC. Esse módulo depende de alimentação externa aplicada em 125 V.

Os sinais de 125 V aplicados na entrada do ME, são isolados através dos optoacopladores (4N29) que têm por objetivo garantir a passagem de informação digital ao sistema central, que no caso deste projeto, será um Raspberry pi.

Em associação com o optoacoplador (ver Figura 3.2), existe um sistema de proteção composto pela associação de resistores de 10 k Ω e um varistor de limite em 250 V. Tal componente se faz necessário para os casos de sobretensão e possui esta característica citada, devido a entrada de energização do circuito estar determinada em 125 V. Na saída do LED emissor do optoacoplador há um capacitor de 33 nF em associação com um diodo que garante a condução de corrente somente em um sentido através da unidade óptica da entrada do CI 4N29. Para determinar o valor de capacitância a ser utilizado neste circuito, foi necessária a análise através do cálculo mostrado nas equações 3.1, 3.2 e 3.3. Na equação 3.1, infere-se a relação das potências de um circuito RC para equacionamento da frequência sobre um capacitor.

$$f = \frac{1}{2 * \pi * R * C} \quad (3.1)$$

A equação 3.2, mostra a relação inversa do cálculo de tensão sobre um circuito simples em malha fechada, baseado pela Lei de Ohm.

$$f = \frac{1}{2 * \pi * \frac{V}{I} * C} \quad (3.2)$$

Por último a equação 3.3, objetiva isolar a variável correspondente ao valor de capacitância do dispositivo relacionado às demais equações apresentadas anteriormente.

$$C = \frac{i}{2 * \pi * f * V} \quad (3.3)$$

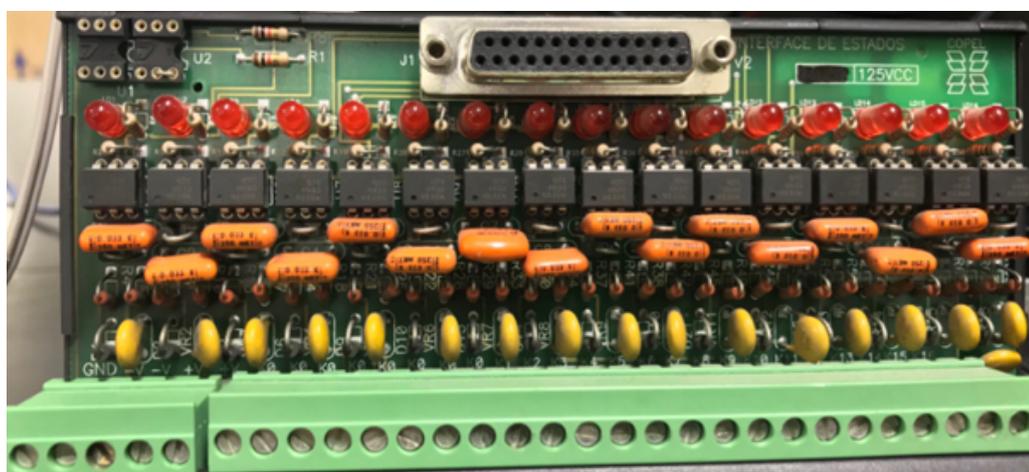
Tendo por definição do sistema os valores de tensão [V] em 125 V, frequência [f] em 60 Hz e corrente [i] em 100 μ A, obteve-se o valor de capacitância apresentado na equação 3.4.

$$C = \frac{100\mu}{2 * \pi * 60 * 125} = 2,12 \text{ nF} \quad (3.4)$$

Uma vez que o valor calculado não é comercialmente encontrado, utilizou-se o valor que mais se aproxima. Para este caso então, determinou-se o valor de capacitância de 33 nF.

O ME completo possui 16 módulos idênticos ao da Figura 3.2, sendo viável portanto receber até dezesseis estados individuais de atuação providos do RTAC.

FIGURA 3.2 – MÓDULO DE ESTADOS UTILIZADO ATUALMENTE PELA COPEL

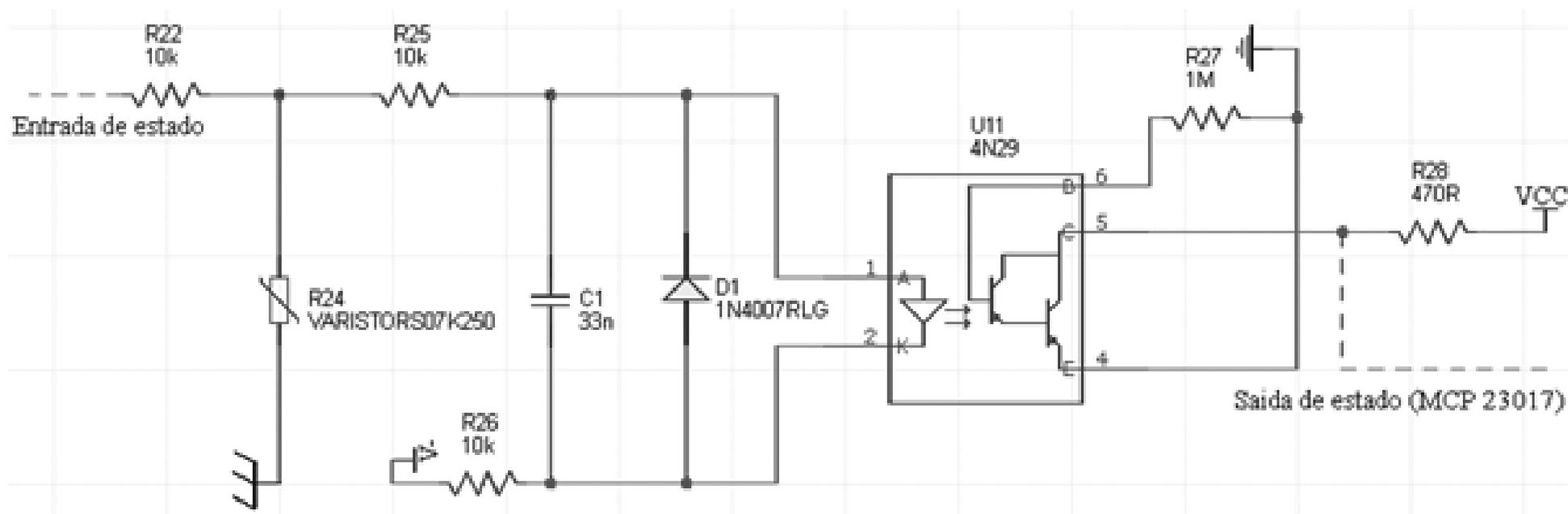


Fonte: Os Autores (2018)

A interface de estados utilizada atualmente pela Copel é exibida na Figura 3.3. Ela possui basicamente os mesmos componentes utilizados na nova interface de estados, só que contém LEDs para indicar o estado das entradas, o que não será mais necessário com o uso do novo ME, devido a visualização dos estados através de software e tela *touchscreen*.

Para integração da interface de estados ao Raspberry Pi foi necessário realizar pequenas atualizações no circuito esquemático, já que a saída calculada no circuito

FIGURA 3.3 – CIRCUITO ESQUEMÁTICO DE APENAS 1 DOS 16 MÓDULOS DE ENTRADA DA NOVA INTERFACE DE ESTADOS PROPOSTA



Fonte: Os Autores (2018)

optoacoplador é de 5 V, responsável por indicar tensão às portas digitais do computador e conferir assim o lançamento de uma carga em uma de suas saídas.

3.1.2 Interface de comandos (MC - Módulo de Comandos)

Os MCs são responsáveis por disponibilizar ao campo, contatos de relés para comandos. Nem sempre os contatos destes relés poderão suportar comando direto sobre os equipamentos, devido a capacidade dos contratos, para condução ou interrupção de corrente. Eles são fornecidos em interfaces de até 8 relés, escolhidos pela necessidade e espaço disponível para instalação dos mesmos. Estes módulos (ver Figuras 3.4) oferecem na saída relés de contato que são acionados com o intuito de gerar comandos a serem assumidos pelos disjuntores das fontes de energia da subestação, como a linha proveniente da rede básica, a do transformador ou a do grupo gerador motor.

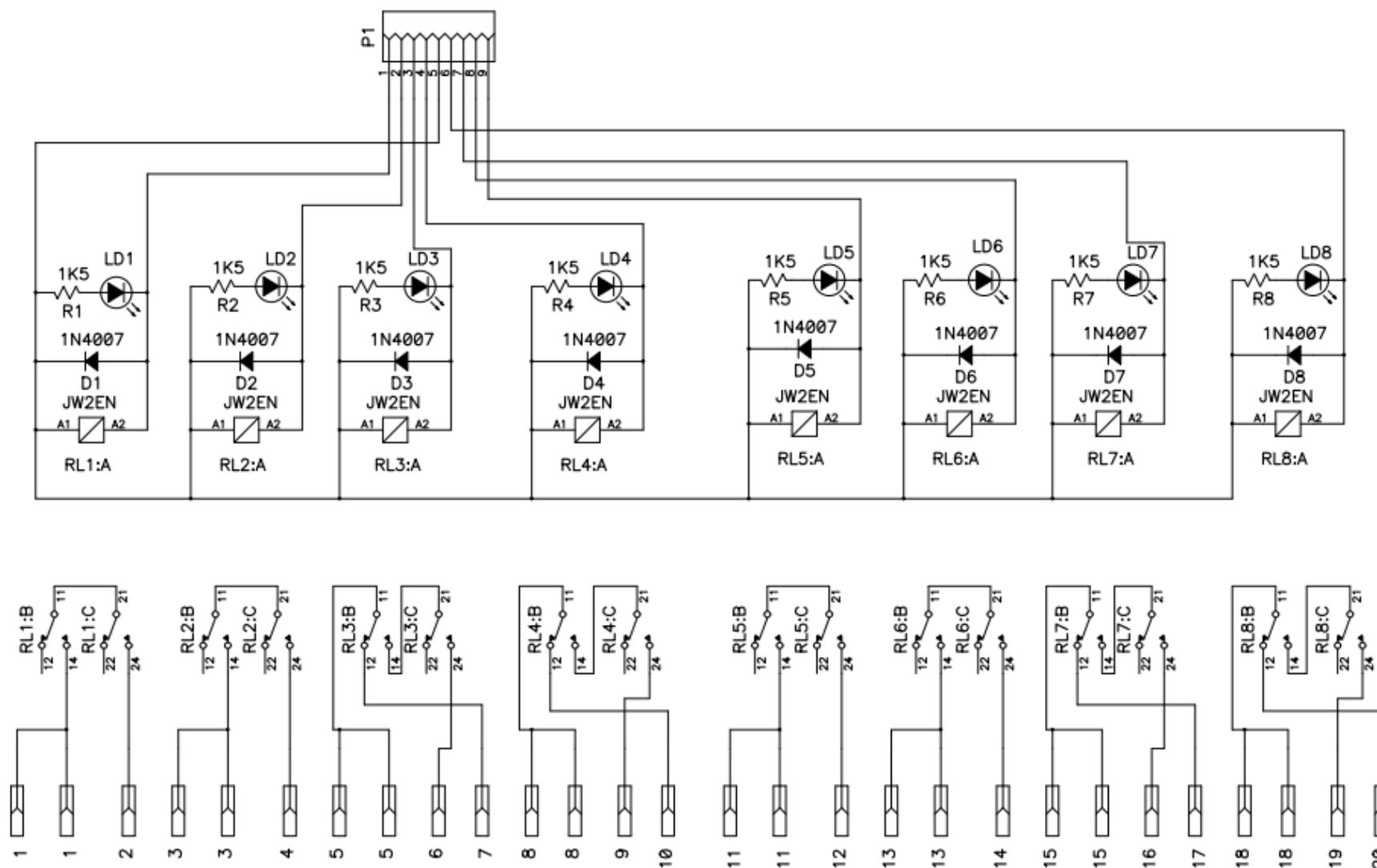
Essas fontes apresentam características específicas que devem ser levadas em consideração por este módulo. Por exemplo, a fonte proveniente da rede básica geralmente é fonte primária de fornecimento de energia à subestação e por isso caso seu disjuntor esteja fechado o sistema deve assumir que somente mais o disjuntor que conecta as barras que alimentam as cargas essenciais e não essenciais esteja fechada.

Todos os outros disjuntores das fontes redundantes devem permanecer abertos para evitar a implicação de curto circuito no sistema e assim possíveis acidentes ou desligamento total da subestação. Outro exemplo seria considerar o tempo de estabilização do grupo motor gerador até seu ponto de atuação, onde a tensão deve estar linear e a frequência do motor a diesel em 60 Hz.

O circuito esquemático do módulo para atuação em um único relé, a exemplo dos módulos de estado, é descrito por uma conexão serial DB9 que conecta 8 relés eletromecânicos. Entre essa conexão existe um resistor em série com um LED como forma de identificar a passagem de corrente na trilha e um diodo de *flyback* que tem a função de proteger o relé contra picos de tensão que podem vir a atingir o circuito, e danificar ou interromper o seu funcionamento.

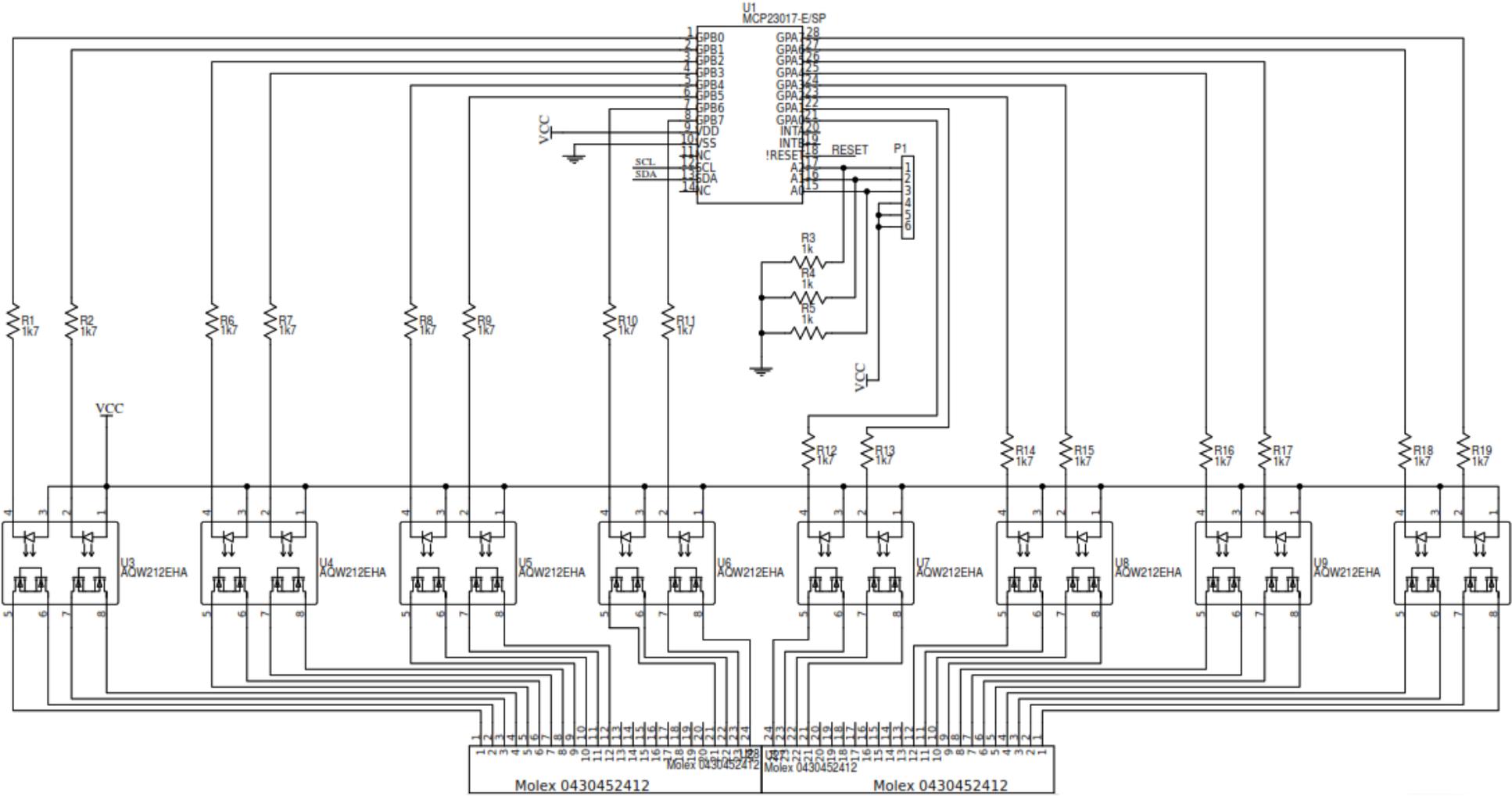
Para implementação do computador embarcado à interface de comando, foram necessárias mudanças no circuito esquemático do MC antigo, devido principalmente

FIGURA 3.4 – CIRCUITO ESQUEMÁTICO DA INTERFACE DE COMANDOS UTILIZADA PELA COPEL ATUALMENTE



Fonte: COPEL (2018)

FIGURA 3.5 – CIRCUITO ESQUEMÁTICO DO NOVO MÓDULO DE COMANDO



Fonte: Os Autores (2018)

à utilização de Raspberry Pi que necessita de respostas imediatas em suas portas lógicas para poder simular períodos reais de comutação entre os disjuntores das fontes da subestação. Para isso, foram substituídos os relés que eram antigamente eletromecânicos, por relés *photomos* AQW21 da Panasonic, que conduzem valores de saída através de MOSFETs, diminuindo assim o tempo de resposta do sistema. Para esta determinação, foi necessário avaliar a tensão e corrente emitidas pelo MCP23017 ao relé photomos, para acendimento dos LEDs e consequente chaveamento dos MOSFETs de sua composição. Visto isso atribuiu-se resistores nas entradas dos relés, como componentes passivos para definir uma queda de tensão em seus terminais e impedir curto circuito no LED. O valor atribuído foi o convencional, 1,7 k Ω .

Outra mudança foi a retirada dos LEDs de sinalização, que não serão necessários, uma vez que todas as entradas de comando serão indicadas na tela TFT conectada ao Raspberry Pi, assim como acontece com o módulo de estados. O novo circuito referente à placa modular de interface de comandos é mostrado na Figura 3.6.

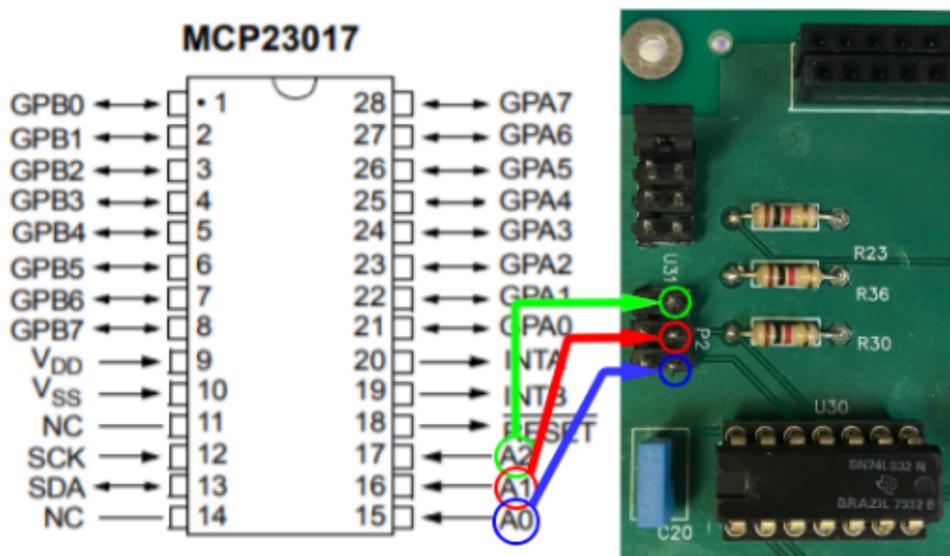
3.2 COMUNICAÇÃO VIA PROTOCOLO I2C

Nas seções anteriores, foram abordados os conceitos e estruturas necessários para o desenvolvimento do hardware e do software que darão funcionalidade ao protótipo. No entanto, para que se possa comunicar a interface de usuário com o controlador em tempo real utilizado na subestação, foi preciso identificar uma maneira de aumentar o número de portas I/O, já que as fornecidas pelo Raspberry Pi não são suficientes para realização de todos os testes dos quais este projeto se propõe.

Após estudo para identificação do melhor método para expandir a quantidade de portas I/O do Raspberry Pi, chegou-se à utilização de um circuito integrado, denominado MCP23017. Este circuito é capaz de agir como um expensor de portas de entradas e saídas múltiplas através de protocolo I2C. Para que seja possível a utilização das portas expandidas de um determinado MCP, o protocolo I2C recebe um endereço característico do dispositivo. Este endereço, representado em 7 bits, é dividido em 4 bits mais significativos já determinados pela sequência "0100", enquanto os outros 3 bits menos significativos devem ser configurados pela inserção de tensão nos endereços A2, A1 e A0 do CI. Esta configuração permite que sejam conectados até 8 circuitos expansores em paralelo ao Raspberry Pi, o que consequentemente gera 128 novas

portas I/O ao todo.

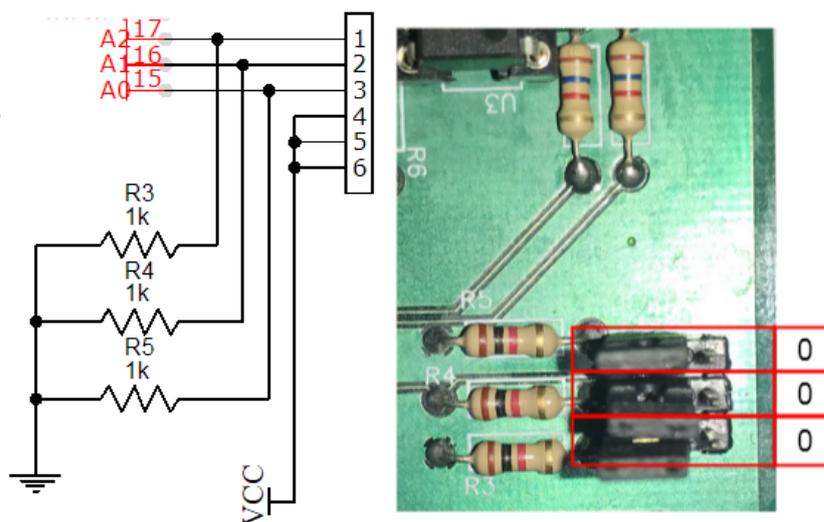
FIGURA 3.6 – ENDEREÇAMENTO DOS MCP23017 PARA IDENTIFICAÇÃO DOS COMANDOS DO PROTOCOLO I2C



Fonte: Os Autores (2018)

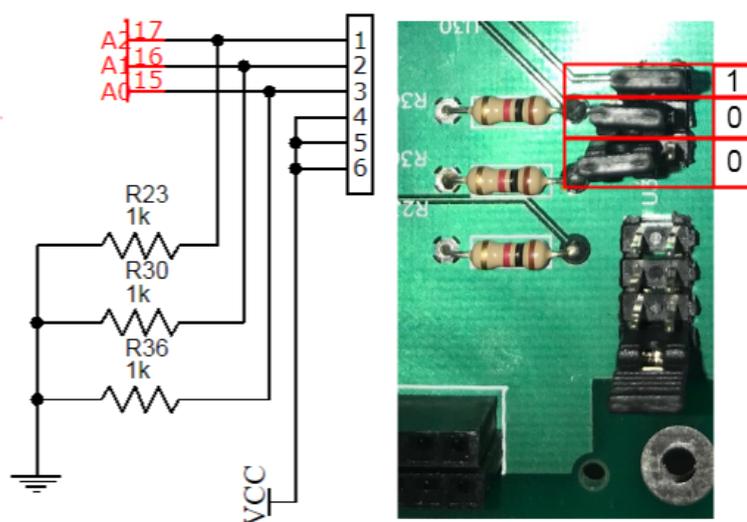
Na Figura 3.6, é possível perceber as referidas portas responsáveis pela representação dos bits menos significativos de endereçamento no MCP, e a referência física aplicada a este projeto representado por um conector 6 pinos no hardware do protótipo. Este conector apresenta 3 dos seus pinos conectados diretamente as portas A2, A1 e A0 do chip, enquanto os outros 3 pinos estão constantemente conectados à 5 V (Vcc). Conforme explicação anterior, o endereço é setado através da inserção de tensão nas portas de endereçamento. Com esta configuração especificada e utilizando-se jumpers de conexão, o endereço é ajustado pela ocorrência de curto entre os conectores. Esta representação é mostrada na Figura 3.7 e na Figura 3.8, que já apresenta o ajuste realizado para caracterizar o endereço do módulo de estados da placa. Neste caso utilizou-se o segundo endereço disponível, ou seja, o valor 0x21 que representa o número binário "0100001". Para o módulo de comando, foi utilizado um outro valor de endereçamento, representado pelo primeiro endereço disponível a ser configurado. Para este caso ajustaram-se todos os jumpers de forma a não gerar curto circuito entre os pinos do conector, e com isso o valor interpretado na entrada do circuito é o 0x20, representado pelo número binário "0100000". Esta configuração pode ser observada através da Figura 3.8.

FIGURA 3.7 – CIRCUITO DE ENDEREÇAMENTO DO MÓDULO DE ESTADOS



Fonte: Os Autores (2018)

FIGURA 3.8 – CIRCUITO DE ENDEREÇAMENTO DO MÓDULO DE COMANDOS



Fonte: Os Autores (2018)

3.3 DESENVOLVIMENTO DO SOFTWARE EMBARCADO

Através da utilização da linguagem de programação Python versão 3.6.5, pôde-se, em um primeiro momento, montar os ambientes gráficos do software de simulação proposto nesse projeto. A biblioteca principal utilizada no desenvolvimento da interface gráfica é a Tkinter, biblioteca essa derivada de duas mais básicas Td e Tcl, que possuem funções gráficas ainda mais elementares. Se trata de uma biblioteca completamente voltada à montagem de estruturas gráficas, específicas para a utilização em sistemas em que existe necessidade de interação homem-software.

A função do software desenvolvido neste trabalho é de basicamente proporcionar ao usuário o controle das simulações que deseja executar no RTAC. Isso quer dizer que o software deve possibilitar o acionamento das portas do MC e monitoramento das portas do ME conectadas ao RTAC. Para que isso seja possível, o usuário deve primeiramente criar ou carregar uma nova configuração de simulação dentro do ambiente do software, etapa essa mostrada mais adiante neste trabalho. Um diagrama completo de caso de uso do software é mostrado na Figura 3.9, sendo todo o detalhamento deste mapa de software descrito nos tópicos que se seguem.

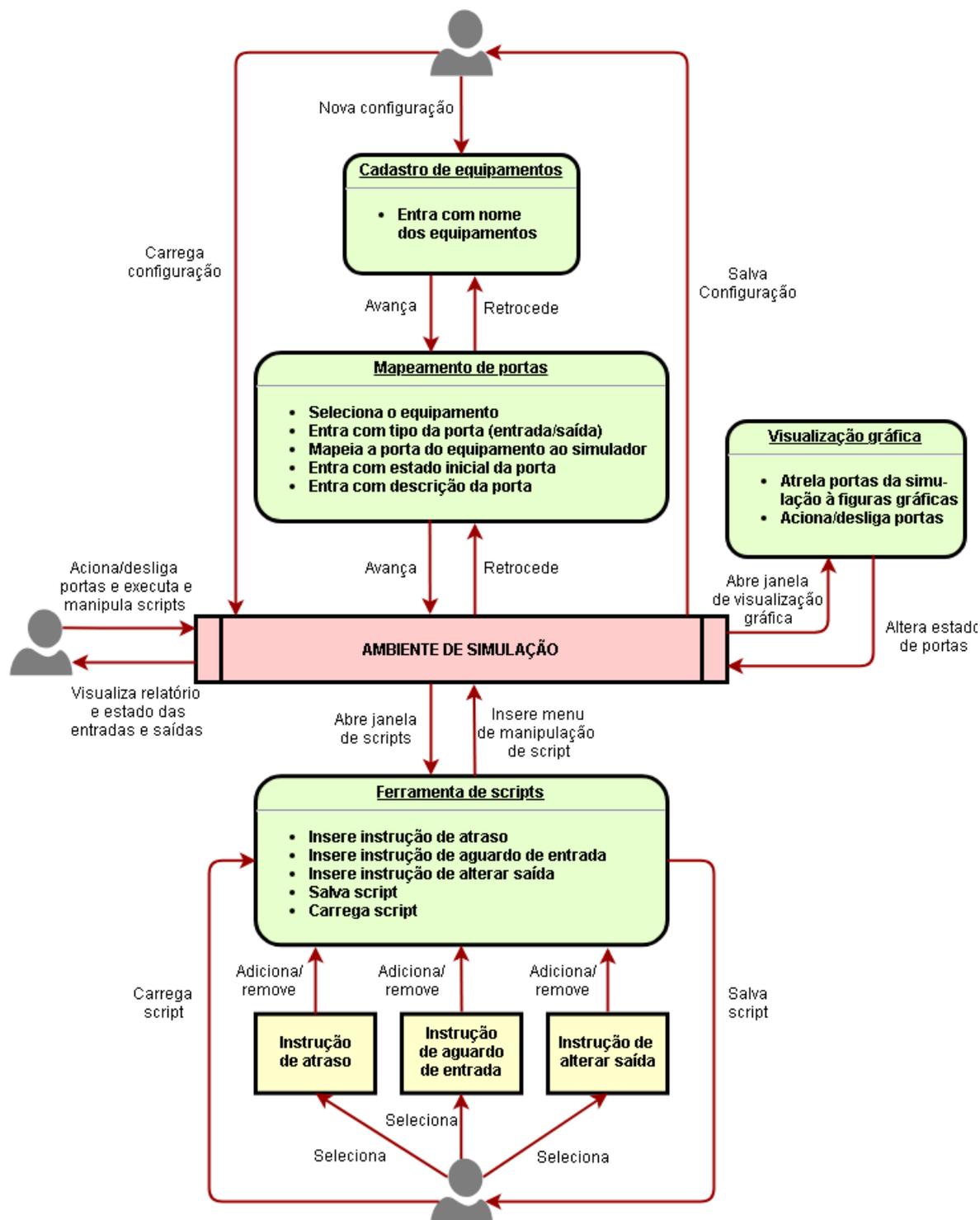
3.3.1 Tela de início e carregamento de configuração

A Figura 3.10 mostra a primeira tela exibida ao usuário no momento em que o software é aberto, sendo possível criar uma nova simulação ou mesmo carregar uma previamente salva. O arquivo de configuração contém a informação a respeito dos equipamentos e mapeamento das portas dos ME e MC que serão utilizadas na simulação em formato de dados JSON. Este arquivo pode ser modificado manualmente por um editor de texto qualquer ou configurado e salvo através do próprio software de simulação. Um exemplo de arquivo de configuração é exibido na Figura 3.11, que possui 3 equipamentos ("GMG", "TSA-1" e "TSA-2") e 3 portas mapeadas a eles com suas respectivas informações necessárias (descrição, número, tipo e estado inicial). Portas do tipo "ENTRADA" necessitam de duas informações extras, que dizem respeito ao tipo de acionamento (*activation type*), se é pulsado ou em nível de estado e, para essa segunda opção (*activation*), se é acionada em nível alto ou baixo, como mostra a configuração da porta de entrada número 1 do equipamento "TSA-2" da Figura 3.11.

3.3.2 Tela de cadastro de equipamentos

No caso de uma nova configuração de simulação, a tela seguinte exibida ao utilizador é a de cadastro de equipamentos. Nela, deverá ser cadastrado o nome dos equipamentos que serão atrelados às portas do MC e ME. Por exemplo, se o usuário deseja simular portas de alarme e estados de um disjuntor dentro da subestação, deverá ser adicionado este equipamento à essa nova simulação, atrelando um nome a ele. A Figura 3.12 exhibe o cadastramento de 6 equipamentos: "TRANSFORMADOR 1", "TRANSFORMADOR 2", "DISJUNTOR A", "RELÉ A1", "DISJUNTOR SACA" e

FIGURA 3.9 – DIAGRAMA DE CASO DE USO COMPLETO DO SOFTWARE SIMULADOR

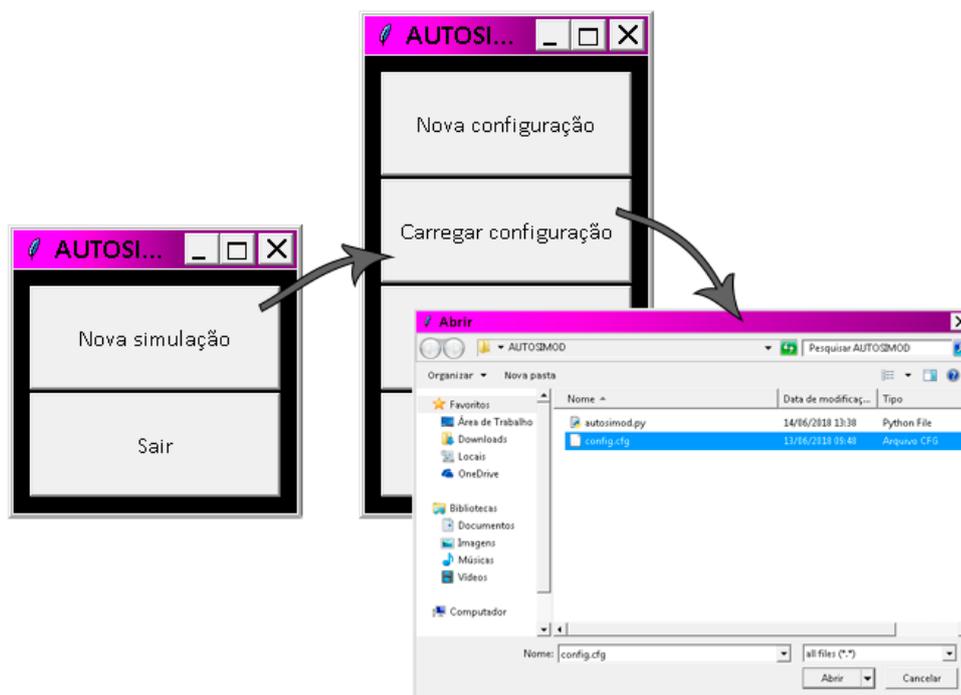


Fonte: Os Autores (2018)

"DISJUNTOR SACC".

Uma próxima etapa essencial na composição de uma nova simulação é o mapeamento de portas. Nesta etapa o usuário irá introduzir portas à simulação, sendo estas associadas aos equipamentos previamente cadastrados, contendo informação

FIGURA 3.10 – TELAS DE INÍCIO E CARREGAMENTO DE ARQUIVO DE CONFIGURAÇÃO DO SOFTWARE DE SIMULAÇÃO



Fonte: Os Autores (2018)

a respeito do tipo da porta, estado inicial, descrição, etc. A tela de mapeamento de portas vem logo após a tela de cadastro de equipamentos.

3.3.3 Tela de mapeamento de portas

Nesta tela o utilizador deve mapear as portas do simulador aos respectivos equipamentos cadastrados. Em geral, o mapeamento de portas necessita de cinco informações relevantes conforme se segue:

- tipo da porta (entrada ou saída);
- nome do equipamento a que a porta pertence;
- número da porta (entradas vão de 0 a 15 e saídas de 16 a 31);
- estado inicial da porta (alto ou baixo)
- tipo do acionamento (pulsado ou estado - exclusivo para entradas)
- nível de acionamento (alto ou baixo - exclusivo para entradas com acionamento do tipo estado)
- descrição da porta (2 a 15 caracteres)

Um exemplo hipotético de cadastramento de portas é exibido na Figura 3.13, onde o utilizador mapeia diversas portas de disjuntores, transformadores e relés (equi-

FIGURA 3.11 – EXEMPLO DE ARQUIVO DE CONFIGURAÇÃO JSON QUE PODE SER CARREGADO NO SOFTWARE DE SIMULAÇÃO, CONTENDO INFORMAÇÃO A RESPEITO DOS EQUIPAMENTOS E PORTAS UTILIZADAS

```
{
  "eqs": [
    "TSA-2",
    "TSA-1",
    "GMG"
  ],
  "ports": [
    {
      "activation": "BAIXO",
      "activation_type": "BAIXO",
      "description": "TENSÃO",
      "equipment": "TSA-1",
      "initial_state": "DESATIVADO",
      "port": "0",
      "port_type": "ENTRADA"
    },
    {
      "activation": "ALTO",
      "activation_type": "ESTADO",
      "description": "TENSÃO",
      "equipment": "TSA-2",
      "initial_state": "ATIVADO",
      "port": 1,
      "port_type": "ENTRADA"
    },
    {
      "description": "PARTIDA-GMG",
      "equipment": "GMG",
      "initial_state": "DESATIVADO",
      "port": "16",
      "port_type": "SAÍDA"
    }
  ]
}
```

Fonte: Os Autores (2018)

pamentos de dentro da subestação) às portas de simulação do software. O usuário deve portanto nomear cada uma das portas desses equipamentos que deseja simular, configurando primeiramente o tipo da porta, se é entrada ou saída, o número da porta que irá usar dentro do simulador e o estado inicial da mesma caso esta seja uma saída.

O simulador possui ao todo 32 portas de simulação, sendo 16 portas de entrada (portas do ME) e 16 portas de saída (portas do MC). As portas de entrada são mapeadas de 0 a 15 e as portas de saídas de 16 à 31 dentro do ambiente de simulação. É importante frisar que uma mesma porta do simulador não pode ser utilizada ao mesmo tempo por outro equipamento, o software restringe o usuário de cometer este tipo de configuração errônea. Após o cadastro dos equipamentos e o consequente mapeamento das portas do simulador às respectivas portas de cada equipamento, o usuário pode então avançar para a tela principal de simulação, onde este terá acesso à tanto ferramentas de visualização quanto de manipulação da simulação, como execução

FIGURA 3.12 – TELA DE CADASTRO DE EQUIPAMENTOS DA SE QUE SERÃO SIMULADOS. O USUÁRIO DEVE INSERIR O NOME DE CADA EQUIPAMENTO



Fonte: Os Autores (2018)

de scripts e artifícios para execuções manuais, alterando o nível lógico das saídas ao toque de um botão sempre que desejar. Os detalhes a respeito da implementação e execução das simulações são passados nos próximos tópicos.

3.3.4 Tela principal de simulação

Exibida na Figura 3.14, a tela principal de simulação possui a visualização completa das entradas e as saídas cadastradas nas telas prévias. Na aba “Entradas”, são exibidas as portas de entrada (leitura dos comandos e estados enviados pelo RTAC) que o usuário mapeou, assim como na aba “Saídas”, que exhibe neste caso as saídas (comandos simulados de alarme e acionamentos enviados ao RTAC), ambas separadas por equipamento e com informação visível em tempo real do estado atual de cada porta, cor vermelha para estado “desabilitada” e cor verde para estado “habilitada”. Ainda nesta tela, a aba “Log” exhibe os acionamentos, estados, comutações e informações gerais a respeito da simulação transcorrida até então, com possibilidade de limpeza sempre que necessitar (“Arquivo” > “Limpar log”). Nas saídas do simulador, o utilizador pode efetuar a comutação individual de cada uma das portas ao toque de um botão, clicando em “LOW” ou “HIGH”.

FIGURA 3.13 – TELA DE MAPEAMENTO DAS PORTAS DO SIMULADOR AOS EQUIPAMENTOS PREVIAMENTE CADASTRADOS



Fonte: Os Autores (2018)

FIGURA 3.14 – TELA PRINCIPAL DE SIMULAÇÃO DO SOFTWARE



Fonte: Os Autores (2018)

A partir da tela principal de simulação, o usuário tem acesso à funcionalidades extras, como visualização gráfica em tempo real dos estados e equipamentos da SE configurados e ainda criação de scripts de simulação, cuja funcionalidade será explicada no próximo tópico. Ainda nesta tela o usuário tem a possibilidade de salvar sua configuração, indo até o menu “Arquivo” e então “Salvar configuração”. Uma

janela será exibida ao usuário para que escolha o local de salvamento do arquivo de configuração. O arquivo poderá ser carregado posteriormente nas telas iniciais do software, exibida na Figura 3.10. Vale citar que sempre que for preciso, o usuário pode clicar nos botões de “Voltar” e “Sair”, presentes em todas as telas do software, o que garante boa usabilidade do mesmo.

3.3.5 Tela de criação de scripts de simulação

Ao clicar no botão “scripts” na tela principal de simulação, o usuário se depara com uma nova janela em que poderá criar scripts de simulação (ver Figura 3.15). scripts de simulação nada mais são que execuções e leituras programadas das saídas e entradas do simulador respectivamente, que podem ser configuradas de modo intuitivo na janela de criação de scripts. A possibilidade de criação e execução de scripts proporciona diversas melhorias na simulação, dentre elas:

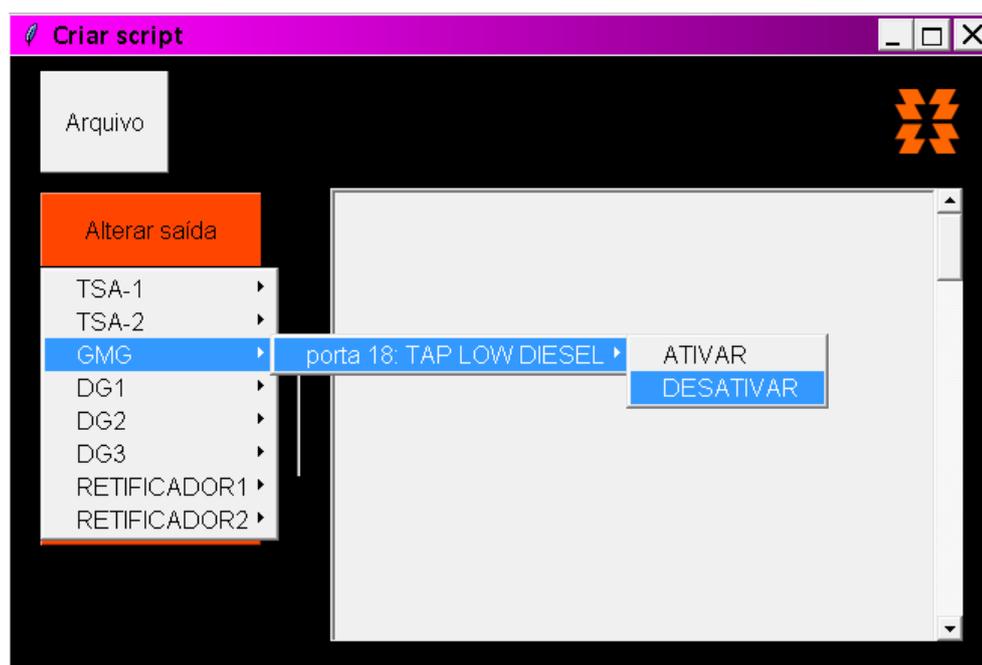
- acionamentos temporizados;
- acionamentos atrelados a comutações da entrada;
- acionamentos sequencias de alta velocidade (humanamente impossível de executar através da tela principal de simulação);
- prevenção de erros de execução manual;
- possibilidade de simular muitos estágios sem perder a confiabilidade da execução;
- possibilidade de salvar/carregar uma sequência de simulação pronta;
- agilidade e robustez na execução da simulação;
- maior confiabilidade nos resultados da simulação;

Conforme exibido nas Figuras 3.16 e 3.17, o usuário pode compor um script utilizando o menu de botões na parte esquerda da janela, em que pode selecionar e adicionar apenas 3 tipos de instruções que irão compor um arquivo de script, são elas:

- alterar saída: altera estado de uma saída configurada na simulação;
- aguardar entrada: aguarda comutação de uma entrada configurada na simulação;
- adicionar atraso: aguarda determinados segundos antes de executar a próxima instrução

Com o encadeamento lógico desses 3 tipos de instruções, é possível automati-

FIGURA 3.15 – TELA DE CRIAÇÃO DE SCRIPTS. EM EVIDÊNCIA A INSERÇÃO DE UMA INSTRUÇÃO DE DESATIVAMENTO DE ALARME “TAP LOW DIESEL” DO EQUIPAMENTO “GMG”



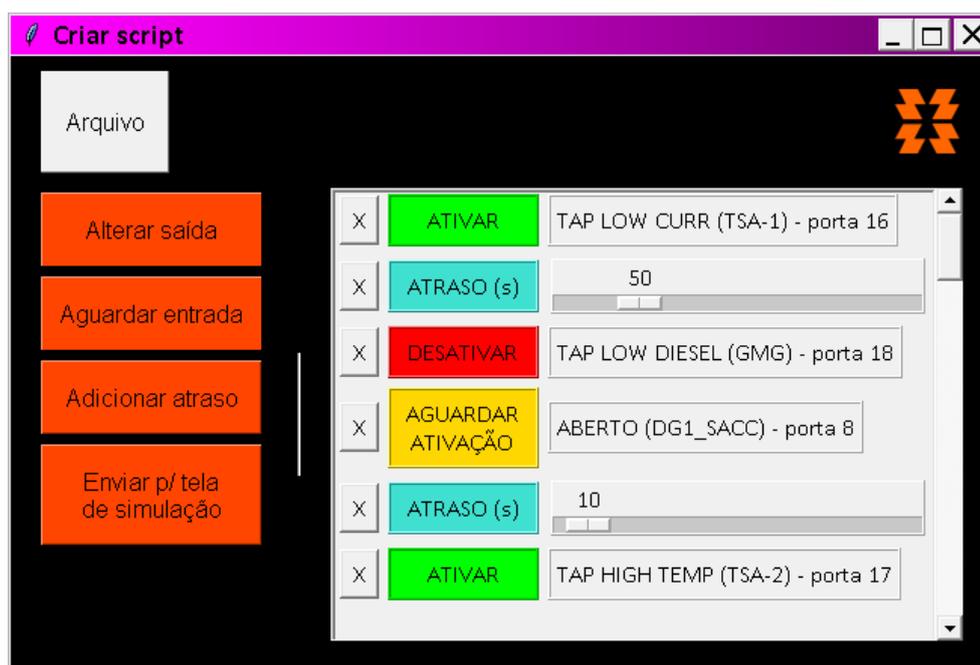
Fonte: Os Autores (2018)

zar completamente uma simulação, restando ao usuário apenas assistir à execução da mesma, visualizando os eventos de comutação das portas tanto na tela principal de simulação, quanto na exibição gráfica, cujos detalhes serão abordados no próximo tópico. A Figura 3.17 apresenta também um script exemplo já montado com 6 instruções a partir de uma configuração prévia de equipamentos e mapeamento de portas das telas anteriores. Conforme mostrado na Figura 3.15, as instruções serão sequencialmente executadas conforme se segue:

1. “ATIVAR” - “TAP LOW CURR (TSA-1)” - porta 16: ativa saída nº 16;
2. “ATRASSO (s)” : aguarda 50 segundos para executar a próxima instrução;
3. “DESATIVAR” - “TAP LOW DIESEL (GMG)” - porta 18: desativa saída nº 18;
4. “AGUARDAR ATIVAÇÃO” - “ABERTO (DG1SACC)” - porta 8: aguarda ativação de entrada nº 8;
5. “ATRASSO (s)” - aguarda 10 segundos para executar próxima instrução;
6. “ATIVAR” - “TAP HIGH TEMP (TSA-2)” - porta 17: ativa saída nº 17

Como é possível observar na Figura 3.17, o usuário pode configurar o tempo de atraso que desejar entre cada instrução, sendo capaz de inserir atrasos que vão de 1 até 250 segundos. Nesta mesma tela, é possível salvar a configuração de script ou

FIGURA 3.16 – TELA DE CRIAÇÃO DE SCRIPTS CONTENDO UM SCRIPT EXEMPLO COM 6 INSTRUÇÕES. À ESQUERDA OS BOTÕES QUE ADICIONAM INSTRUÇÕES, À DIREITA AS INSTRUÇÕES JÁ ADICIONADAS



Fonte: Os Autores (2018)

mesmo carregar um script previamente montado, bastando acessar o menu “Arquivo” e então “Carregar script” ou “Salvar script”. scripts de simulação são salvos em formato JSON assim como as configurações de equipamentos e mapeamento de portas. O script da Figura 3.17 salvo em formato JSON é exibido na Figura 3.18.

Havendo terminado a construção do script, o usuário então pode executá-lo clicando no botão “Enviar p/ tela de simulação”. Ao fazer isto, um novo menu de execução de scripts é exibido na tela principal de simulação conforme mostra a Figura 3.18. Através desse menu, o usuário pode controlar a execução das instruções do script, podendo pausa-lo, voltar ou mesmo avançar passos de execução por meio dos botões deste menu. O software avisa o utilizador sobre o encerramento do script, assim como na inicialização. Fica muito prático para o usuário avançar passo a passo a simulação carregada, executar toda ela ou mesmo executá-la a partir de um ponto específico no meio do script. Na tela principal de simulação o utilizador consegue ao mesmo tempo acompanhar o relatório de eventos deflagrados pelo script e visualizar a comutação das portas de entrada e saída em tempo real.

FIGURA 3.17 – CONFIGURAÇÃO SALVA DO SCRIPT DA FIGURA 3.17 EM FORMATO JSON

```

{
  "script": [
    {
      "description": "TAP LOW CURR",
      "equipment": "TSA-1",
      "port": "16",
      "type": "ATIVAR"
    },
    {
      "type": "ATRASO",
      "value": 50
    },
    {
      "description": "TAP LOW DIESEL",
      "equipment": "GMG",
      "port": "18",
      "type": "DESATIVAR"
    },
    {
      "description": "ABERTO",
      "equipment": "DGL_SACC",
      "port": "8",
      "type": "AGUARDAR\\nATIVAÇÃO"
    },
    {
      "type": "ATRASO",
      "value": 10
    },
    {
      "description": "TAP HIGH TEMP",
      "equipment": "TSA-2",
      "port": "17",
      "type": "ATIVAR"
    }
  ]
}

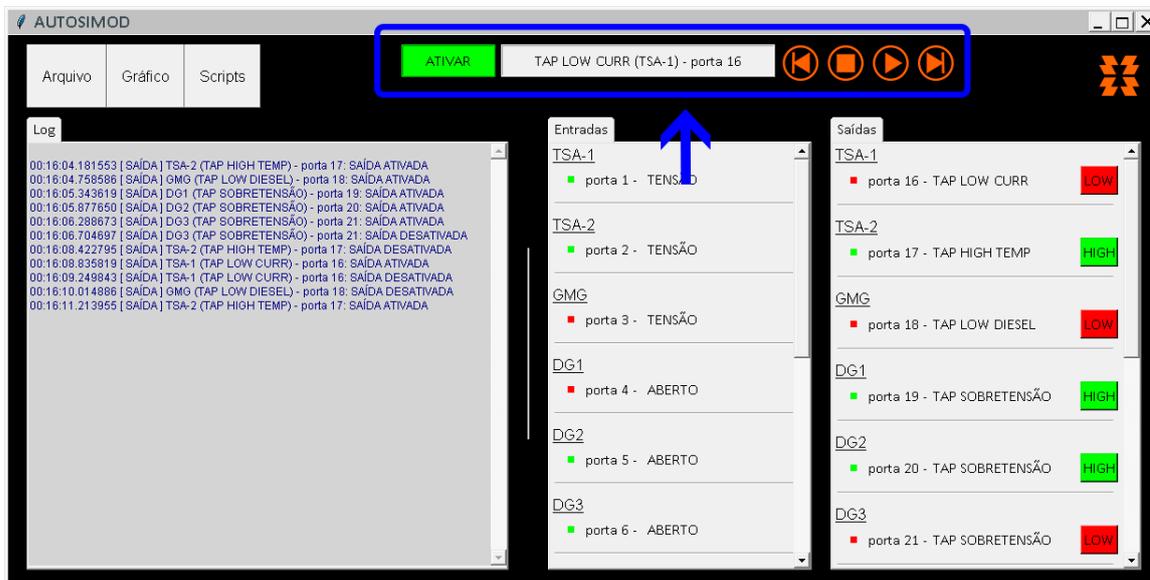
```

Fonte: Os Autores (2018)

3.3.6 Telas de exibição gráfica da SE

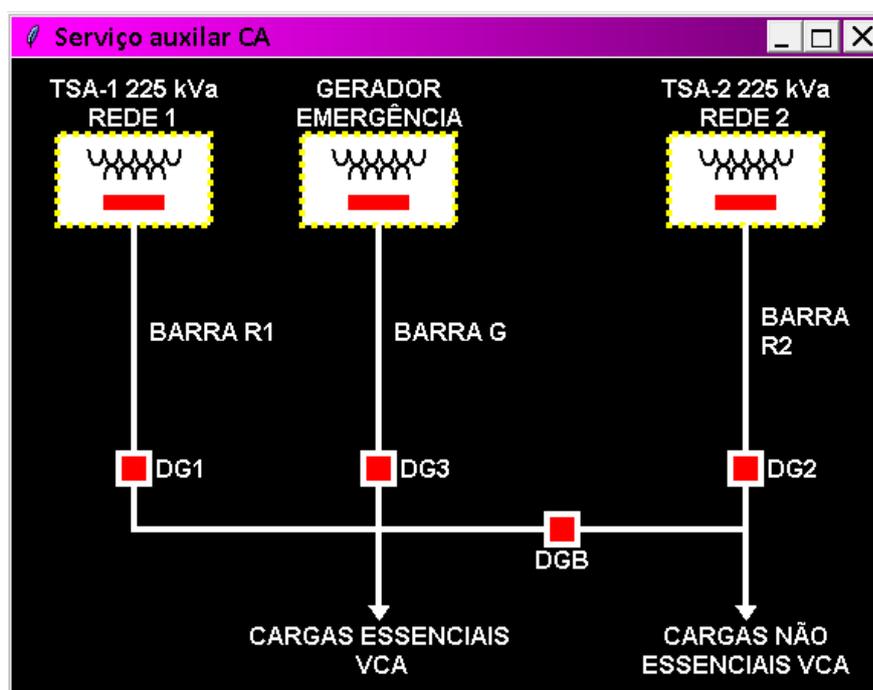
De modo a proporcionar melhor visualização dos cenários de simulação, foram construídos dois modelos gráficos especialmente desenvolvidos para a simulação dos SAs da subestação CTN. Nos modelos gráficos em questão, o usuário pode cadastrar, acionar e verificar visualmente e em tempo real os estados e sinais dos principais equipamentos que compõe os SACC e SACA da CTN. A Figura 3.19 exhibe o modelo gráfico unifilar do SACA da CTN, seguindo exatamente o mesmo padrão visual conforme encontra-se na Figura 2.3. É possível observar na Figura 3.20 que facilmente o usuário consegue atrelar uma porta de simulação à um equipamento da tela gráfica, que no caso evidenciado da imagem, está sendo atrelado ao disjuntor “DG3” sua respectiva porta de alarme de sobretensão, ou seja, toda vez que o usuário comutar essa porta, seja por script, ou via manual, nesta mesma tela poderá ser visto a comutação ocorrendo em tempo real. Telas gráficas como as descritas anteriormente, podem ser adicionadas ao software conforme a demanda de novos projetos e simulações, sendo a solução proposta por este trabalho uma concepção universal, suportando qualquer tipo de simulação digital conforme a necessidade dos usuários.

FIGURA 3.18 – TELA PRINCIPAL DE SIMULAÇÃO COM EVIDENCIAÇÃO DO MENU DE EXECUÇÃO DE SCRIPTS



Fonte: Os Autores (2018)

FIGURA 3.19 – TELA DE EXIBIÇÃO GRÁFICA ESPECIALMENTE DESENVOLVIDA PARA SIMULAÇÃO DO SACA DA CTN. ENTRADAS E SAÍDAS PODEM SER ATRELADAS AOS ELEMENTOS GRÁFICOS



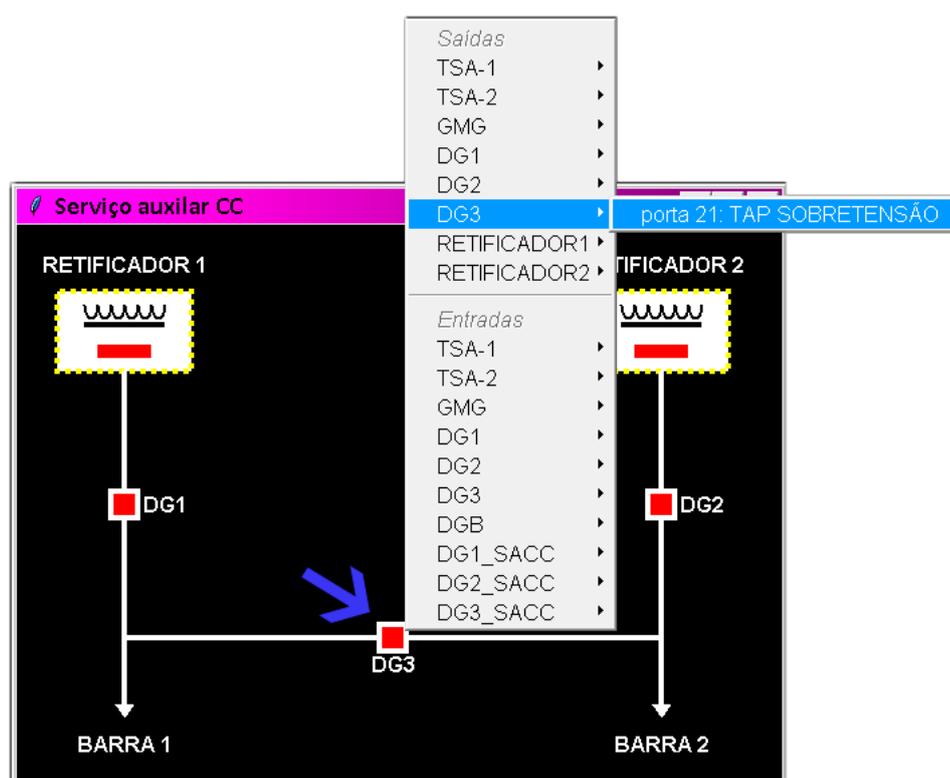
Fonte: Os Autores (2018)

3.3.7 Interação hardware-software

Nos tópicos anteriores foram exibidas as interfaces de criação e controle de simulação disponíveis ao usuário dentro do ambiente de software. No entanto, ainda

deve ser esclarecido como ocorre o diálogo entre essa interface humana e o hardware deste projeto. O protocolo I2C faz a interface entre os módulos de comando e estado com o software embarcado no RPi. É ele quem governa a passagem dos eventos enviados pelo RPi para o ME e a passagem de eventos recebidos pelo MC. Todo evento enviado pelo RTAC ao simulador ou evento enviado do simulador ao RTAC, segue o caminho descrito na Figura 3.21, onde é mostrado o mapeamento das portas do software nas conexões existentes entre os módulos e o RTAC. Entende-se por evento a comutação de uma porta, seja para nível alto ou para nível baixo.

FIGURA 3.20 – TELA DE EXIBIÇÃO GRÁFICA ESPECIALMENTE DESENVOLVIDA PARA SIMULAÇÃO DO SACC DA CTN, SENDO EVIDENCIADO O ATRELAMENTO DA SAÍDA DE SOBRETENSÃO DO DISJUNTOR “DG3” AO ELEMENTO GRÁFICO QUE O REPRESENTA



Fonte: Os Autores (2018)

O protocolo I2C é inicializado logo após o usuário finalizar o mapeamento de portas no software, descrito no tópico 3.3.3. No momento em que a tela principal de simulação é apresentada ao usuário, uma rotina é executada pelo software. Essa rotina basicamente realiza um série de configurações inicialmente nas portas GPIO do RPi e posteriormente nos MCPs dos módulos de comando e estados. As configurações são executadas conforme a seguinte lista ordenada:

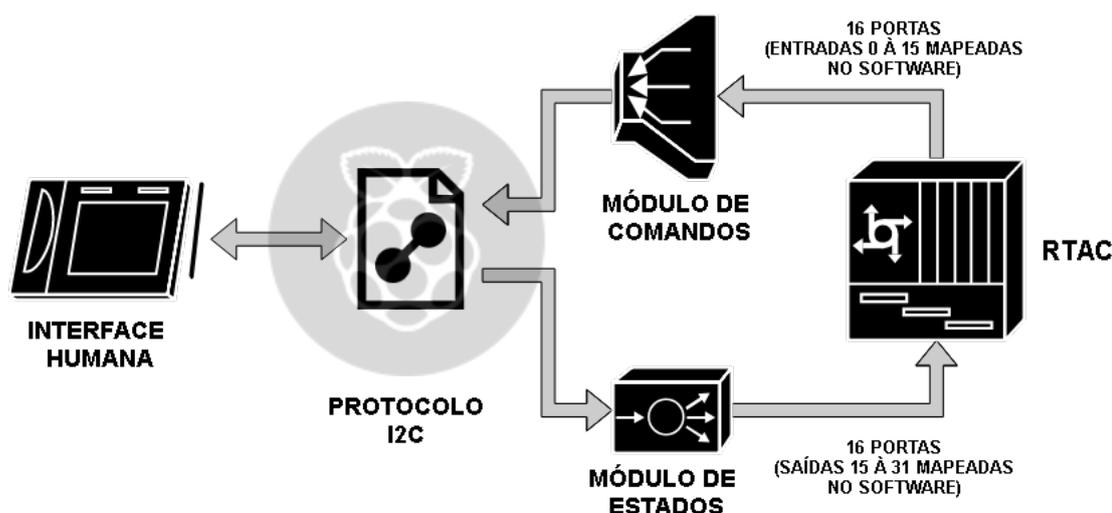
1. configura porta GPIO 6 do RPi como entrada, nível padrão “baixo” e habilita uma interrupção para o caso dessa porta ser acionada em nível alto. Essa porta se trata da porta de sinal IO31 exibida na Figura 3.23;

2. configura portas de A0 à A7 e B0 à B7 como entradas com pull-up em nível alto e interrupções ativas para todas elas no MCP do módulo de comandos;

3. configura portas de A0 à A7 e B0 à B7 como saídas com pull-up em nível alto no MCP do módulo de estados;

4. carrega nas saídas os estados predefinidos na tela de mapeamento de portas do tópico 3.3.3. O código implementado para execução desta rotina de inicialização está sendo exibido devidamente comentado na Figura 3.23.

FIGURA 3.21 – INTERAÇÃO ENTRE SOFTWARE E HARDWARE. O PROTOCOLO I2C INTERMEDIA OS COMANDOS RECEBIDOS PELO RTAC E OS ESTADOS ENVIADOS A ELE ATRAVÉS DO SOFTWARE



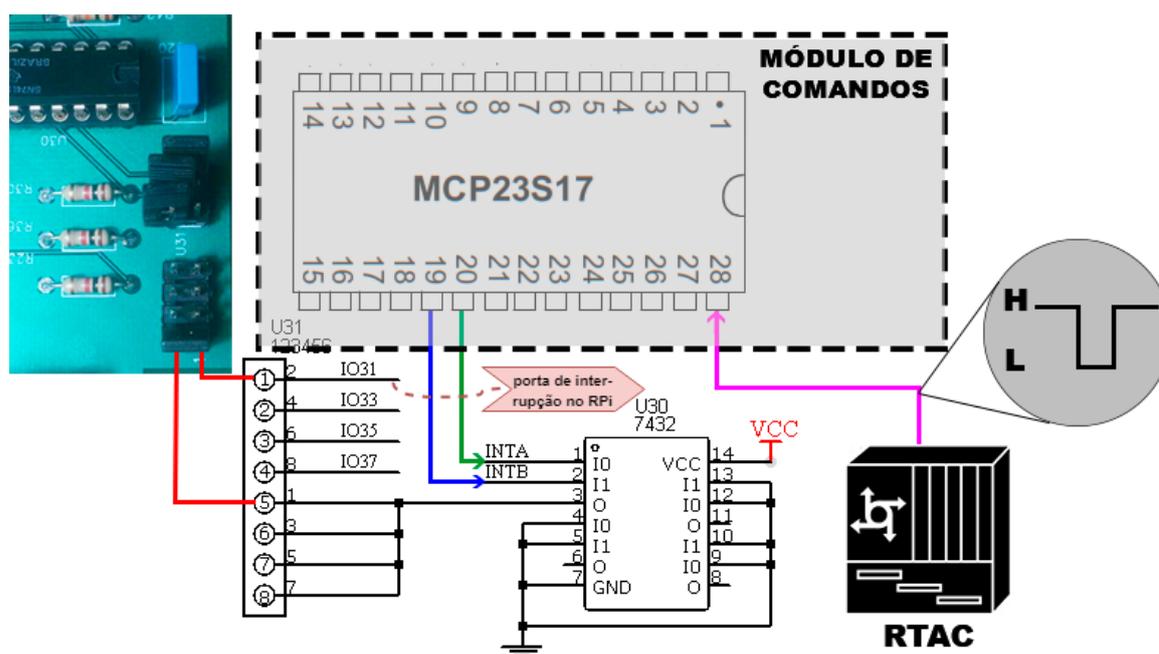
Fonte: Os Autores (2018)

As leituras vindas do RTAC são pulsadas ou de nível de estado lógico. Para o tipo pulsado, tanto as comutações para nível alto quanto as comutações para nível baixo não mantêm a tensão nesses níveis. Para tanto, é necessário que haja uma forma de memorizar a informação de comutação de uma porta do RTAC, para que o módulo de comandos possa avisar ao software e por fim ao utilizador, que um evento foi recebido em determinada porta. Portanto, para tratar esse tipo de sinal pulsado de forma adequada, foram utilizadas as entradas de interrupção do CI MCP23017.

Ao comutar uma porta, o RTAC envia essa informação de forma pulsada para o módulo de comandos (acionamento em nível baixo) e este disponibiliza esse

sinal na porta correspondente no CI MCP23017. Quando o CI recebe este pulso, automaticamente aciona suas duas interrupções “INTA” e “INTB” em nível alto. Essa informação passa por uma porta do tipo E que então aciona a sua saída (porta 3) em nível alto. Essa informação é então repassada através da seleção de pinos na própria placa, que neste caso está setada para enviar a interrupção para a porta IO31 no RPi. Este pino é monitorado continuamente pelo software que após dada a interrupção, lê o registradores INTFA e INTFB que contém a informação de qual porta do MCP23017 ocasionou esta interrupção. Finalmente, o software disponibiliza o informe da comutação da entrada ao usuário via ambiente gráfico. A Figura 3.22 mostra o esquema descrito.

FIGURA 3.22 – INTERPRETAÇÃO DAS INTERRUPTÕES ACIONADAS PELO CI MCP23017 NO MÓDULO DE COMANDOS, AO RECEBER UM EVENTO DE COMUTAÇÃO DE PORTA VINDA DO RTAC EM FORMA DE PULSO NA PORTA 28



Fonte: Os Autores (2018)

Cada porta de entrada possui proteção *anti-bouncing* individualizada via software. Essa proteção faz com que interrupções providas de uma mesma porta dentro de uma janela menor do que 3 milisegundos sejam descartadas. Isso quer dizer que diversas portas de entrada podem receber eventos praticamente ao mesmo tempo, mas que individualmente, por cada porta, somente são tratados eventos espaçados de 3 em 3 milisegundos. Caso fosse adotada um proteção *anti-bouncing* em nível de hardware, todas as portas do MCP23017 do módulo de entradas ficariam submetidas à

mesma janela de tempo citada para cada vez que ocorresse uma interrupção individual. Portanto, a proteção via software utilizada ao invés da de hardware, aumenta a sensibilidade e confiabilidade das leituras do ME.

Os estados enviados ao RTAC seguem uma rotina diferente das leituras feitas do mesmo. Ao comutar uma porta de saída seja via script ou manualmente, a rotina da Figura 3.23 é executada. Essa rotina recebe o número da porta a ser comutada e o estado a ser aplicado. Primeiramente, é feita a leitura dos estados atuais das 16 saídas, e, logo após, a informação de comutação da porta é incorporada ao estado das saídas realizando operação do tipo OU para o caso de o novo estado ser nível lógico baixo ou do tipo E para o caso de ser nível lógico alto. Por fim, o resultado dessa operação lógica é repassada ao barramento I2C que atualiza as saídas do MC. No capítulo de resultados deste projeto será possível observar uma demonstração prática deste funcionamento.

FIGURA 3.23 – ROTINA DE COMUTAÇÃO DE PORTAS DE SAÍDA DO ME. A FUNÇÃO “WRITE-OUTPUT” RECEBE COMO ARGUMENTOS A PORTA QUE SERÁ COMUTADA E O NOVO ESTADO DA MESMA.

```
def write_output(self, port=None, state=None):
    self.portStateOut[port].set(1) if state else self.portStateOut[port].set(0)
    # -----
    # realiza leitura das saídas antes de realizar escrita
    #portas de saída podem ser:
    #A7_A0 (31-24)
    #B7_B0 (23-16)
    output_A = BUS.read_byte_data(ADDRESS_MCP_OUTPUTS, GPIOA)
    output_B = BUS.read_byte_data(ADDRESS_MCP_OUTPUTS, GPIOB)
    mask = 1 << port-16
    if not state: #desligar saída
        A7_A0 = mask >> 8
        B7_B0 = mask & 255
        A7_A0 = output_A | A7_A0
        B7_B0 = output_B | B7_B0
    else: #setar saída
        A7_A0 = 0xFF ^ (mask >> 8)
        B7_B0 = 0xFF ^ (mask & 255)
        A7_A0 = output_A & A7_A0
        B7_B0 = output_B & B7_B0
    BUS.write_byte_data(ADDRESS_MCP_OUTPUTS, OLATA, A7_A0) #escreve na saída
    BUS.write_byte_data(ADDRESS_MCP_OUTPUTS, OLATB, B7_B0) #escreve na saída
```

Fonte: Os Autores (2018)

4 TESTES E RESULTADOS

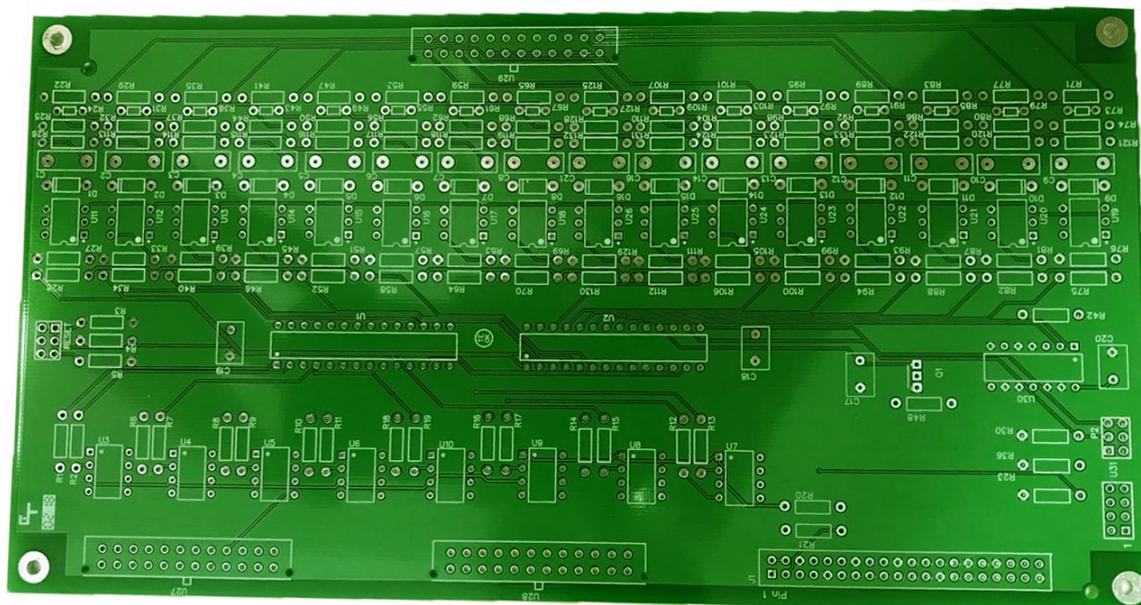
Os esforços realizados na construção e desenvolvimento do simulador, tanto no quesito de software quanto no de hardware, tem se mostrado promissores até então. A programação de uma interface flexível e intuitiva ao usuário, permite que este possa configurar as portas do simulador ao seu modo. A criação e execução de simulações se torna cada vez mais facilitada com a possibilidade de carregamento de configuração prévia e execução de scripts. A constante melhoria da interface gráfica com o usuário é um ponto chave no desenvolvimento do mesmo. A tela principal de simulação mostrada na Figura 3.14 possui uma aba em que o utilizador pode facilmente organizar o espaço do modo que desejar, aumentando ou diminuindo a tela de relatório ou as telas de exibição e manipulação de estado das portas. Portanto, a vista do usuário, o software concebido até e então é altamente intuitivo, flexível e de fácil utilização, contendo as ferramentas necessárias para realização dos ensaios das lógicas de automatismo dos SAs das subestações.

4.1 TESTES DE HARDWARE

Os diagramas esquemáticos apresentados no decorrer deste trabalho, sendo diversas vezes revisados a fim de evitar erros de projeto no circuito, resultaram na confecção da placa PCB da Figura 4.1. O resultado dos trabalhos a cerca do hardware do projeto pode ser visto na Figura 4.2, em que está sendo mostrada uma foto desta mesma placa única que contém as 16 entradas de estado e 16 saídas de comando que serão conectadas ao RTAC. Para cada uma das 16 saídas do MC e 16 entradas do ME, foram efetuados os testes de funcionamento em que basicamente foi ensaiado a transmissão dos sinais digitais para cada uma das 32 portas. Os ensaios incluíram execução de scripts e manipulação através de interface gráfica. Esses ensaios são descritos na seção de integração hardware-software.

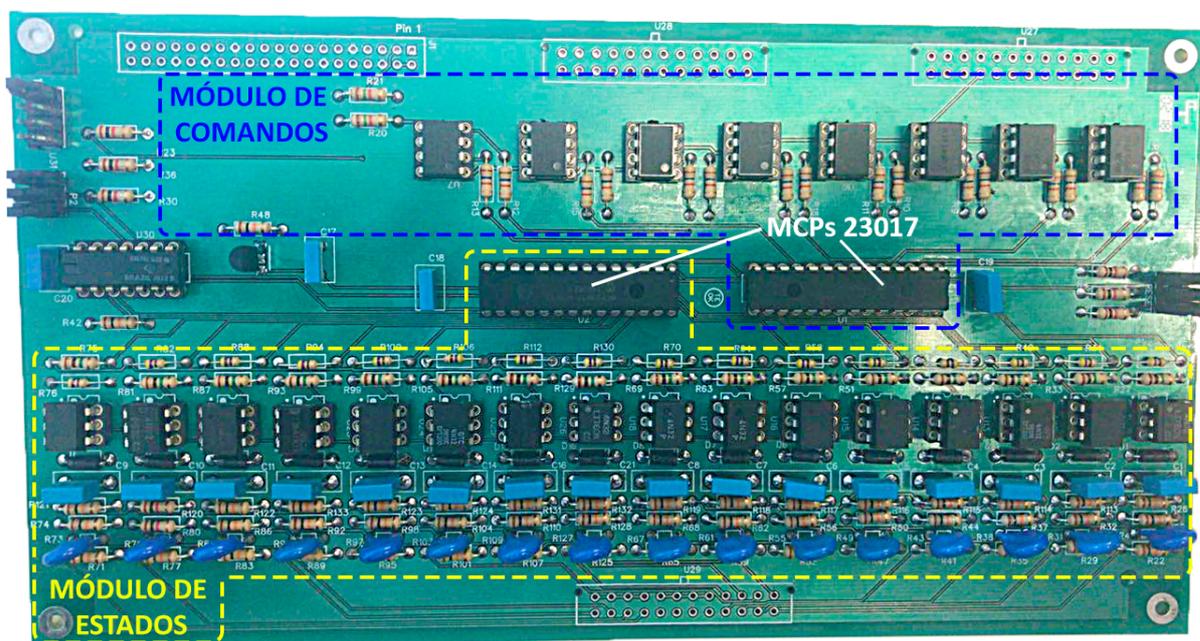
Além disso foi realizado um teste de sincronização com o controlador de automação em tempo real para verificar o funcionamento das respostas dos módulos de comando e de estados. Essa verificação foi realizada através da leitura das interrupções nos MCP23017, que são demonstradas nas imagens do osciloscópio na Figura 4.3.

FIGURA 4.1 – PLACA PCB DO MÓDULO EXPANSOR DE PORTAS



Fonte: Os Autores (2018)

FIGURA 4.2 – PLACA PCB DO MÓDULO EXPANSOR DE PORTAS JÁ COM OS COMPONENTES SOLDADOS. EM EVIDÊNCIA OS MÓDULOS DE ESTADOS E COMANDOS, QUE CONTÉM AS PORTAS DIGITAIS A SEREM CONECTADOS NO RTAC



Fonte: Os Autores (2018)

4.2 TESTES DE SOFTWARE

A interface gráfica do software concebida até então, garante ao usuário o controle total das portas do simulador através de configuração fácil, como já foi mostrado

na seção 3.3. A interface gráfica foi experimentada em todos os cenários possíveis e condições de uso permitidas a fim de encontrar erros de continuidade na programação. Ainda na seção 3.3, são exibidas as telas dos diversos contextos e funcionalidades do software testadas, tendo sido eliminados erros prematuros e erros de continuidade.

4.3 INTEGRAÇÃO HARDWARE-SOFTWARE

Uma vez que o hardware e o software deste projeto atingiram o nível de confiabilidade esperado (após realização dos testes descritos nos tópicos 4.1 e 4.2), ou seja, a comunicação entre ambos se configurou ininterrupta e imediata (tempo real), a integração destes dois elementos do projeto se fez necessária. Essa agregação foi implementada através da utilização do protocolo I2C. Para examinar o funcionamento deste protocolo intermediando a conversação entre software e hardware, foram elaborados dois protocolos de teste, um para o teste das entradas do MC e outro para o teste das saídas do ME, conforme explanado nos tópicos que se seguem. Na Figura 4.3 é possível verificar a integração do protótipo desenvolvido, junto ao controlador (RTAC), conectados a um osciloscópio para validação das leituras e comandos recebidos no hardware. Verificou-se assim que todas as portas digitais estão funcionando em tempo real e que as respostas no software estão em acordo com as variações de sinal lógico, ou seja, estão sendo acionadas através de sinais em borda de subida e de descida.

FIGURA 4.3 – TESTE DE BANCADA DAS RESPOSTAS DO RTAC JUNTO AO HARDWARE DO PROTÓTIPO

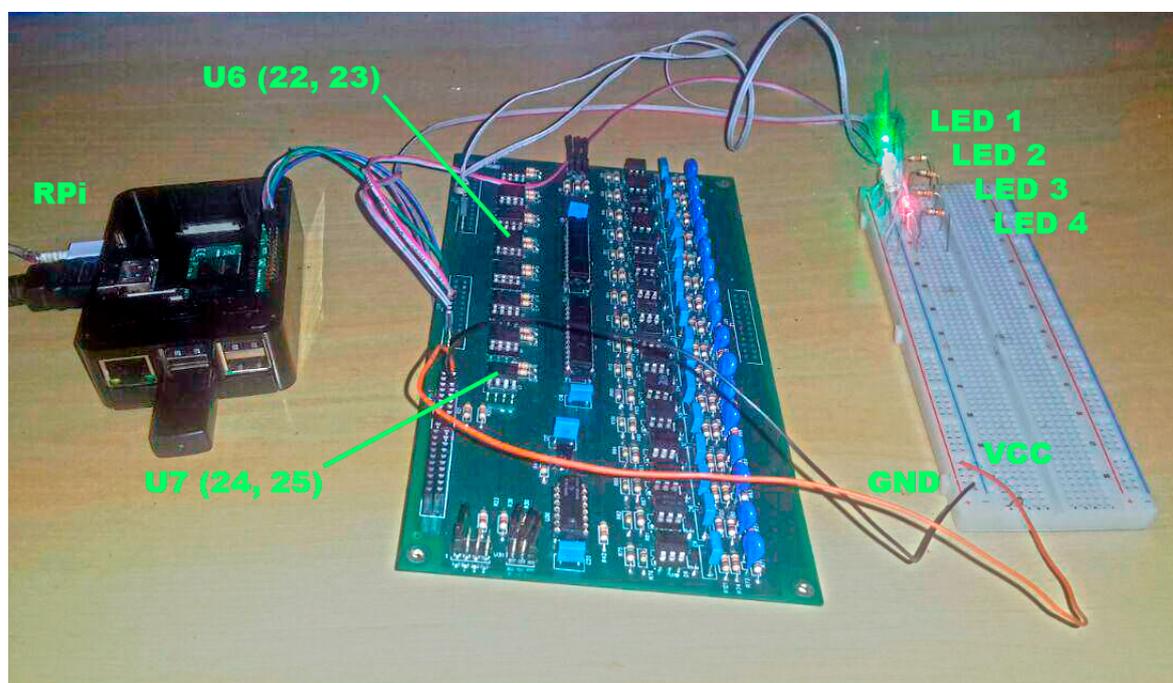


Fonte: Os Autores (2019)

4.3.1 Teste de funcionamento das saídas do ME

Um teste de bancada foi montado conforme mostra a Figura 4.5. Nesta figura é possível observar 4 LEDs em um *protoboard* em que o acionamento de cada um deles está condicionado ao estado de 4 portas arbitrárias do ME (22, 23, 24 e 25), portas essas que serão portanto acionadas/desligadas via software com este teste. A alimentação 5 V e a massa do circuito são providos pelas portas 2 e 6 do RPi respectivamente. A montagem física é exibida na Figura 4.4 e evidencia os componentes envolvidos neste ensaio. Um script para ascender os LEDs 1, 2, 3 e 4 de maneira sequencial com intervalos de 1 segundo entre cada ascendimento, foi montado via interface gráfica. A Figura 4.6 exibe este script. O mesmo foi enviado à tela principal de simulação onde foi executado, obtendo um resultado final exitoso em que todas as instruções foram efetuadas corretamente no tempo em que foram programadas. O relatório desta simulação é exibido na Figura 4.7.

FIGURA 4.4 – MONTAGEM FÍSICA DO TESTE DE BANCADA REALIZADO COM TODAS AS 16 PORTAS DE SAÍDA (DE 4 EM 4). EM EVIDÊNCIA OS COMPONENTES ENVOLVIDOS NO ENSAIO

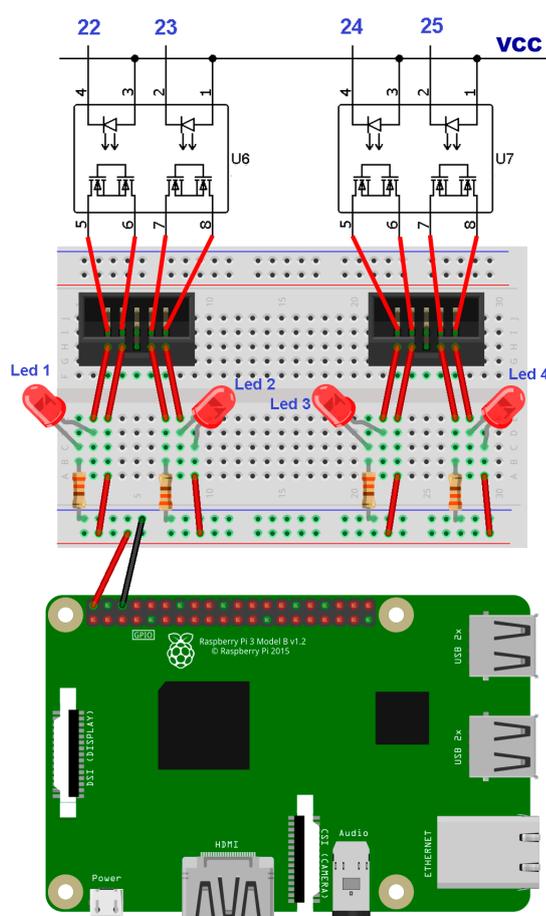


Fonte: Os Autores (2018)

Com o teste de bancada em questão, ficou evidente o correto funcionamento das lógicas de controle do ME (saídas) implementadas via protocolo I2C pelo software. No entanto, ainda é necessário realizar o teste do MC (entradas) da placa expansora,

comutando logicamente o estado de cada uma de suas 16 portas, verificando a interpretação do software perante aos tipos de sinais inseridos (pulsados ou de nível lógico) e com isso, avaliando o correto funcionamento deste módulo. Isso será abordado e descrito no próximo tópico.

FIGURA 4.5 – TESTE DE BANCADA REALIZADO (ANÁLISE DO FUNCIONAMENTO DO PROTOCOLO I2C NA COMUNICAÇÃO COM O ME). AS PORTAS 22, 23, 24 E 25 DO ME CONTROLAM O ASCENDIMENTO DOS LEDS 1, 2, 3 E 4 RESPECTIVAMENTE

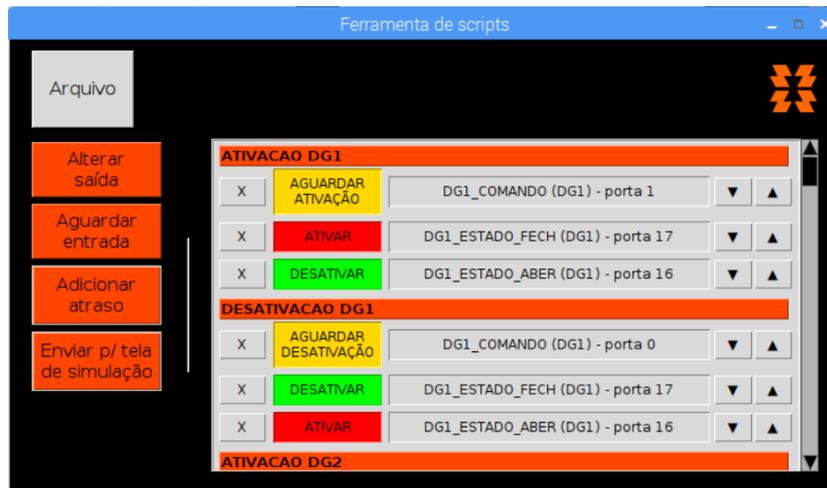


Fonte: Os Autores (2018)

4.3.2 Teste de funcionamento das entradas do MC

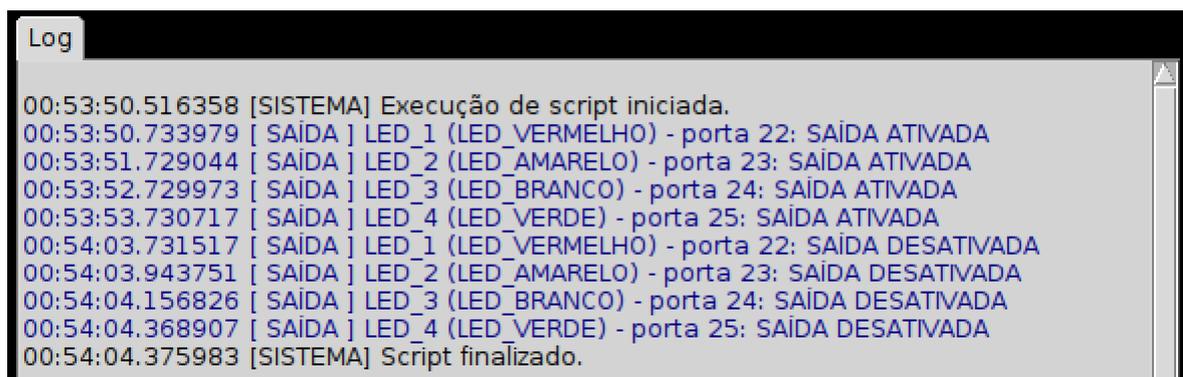
Esses testes também foram realizados em uma bancada e avaliaram o funcionamento das lógicas de interrupção acionadas pelo CI MCP23017 do ME e a rotina de comutação de entradas explicada no tópico 3.3.7 e exibida também na Figura 3.22. A forma de avaliar o funcionamento das portas de entrada compôs primeiramente a configuração das mesmas dentro do software de simulação, conforme mostra a Figura 4.8. Após feito isso, as 16 portas de entrada foram testadas manualmente, setando-se

FIGURA 4.6 – SCRIPT DE TESTE DO ME



Fonte: Os Autores (2018)

FIGURA 4.7 – RELATÓRIO DO TESTE DO ME. OS LEDs ASCENDEM SEQUENCIALMENTE EM INTERVALOS DE 1 SEGUNDO E POR FIM TODOS SÃO DESLIGADOS AO MESMO TEMPO APÓS 10 SEGUNDOS CONFORME INSTRUÇÕES DO SCRIPT



Fonte: Os Autores (2018)

o nível lógico de acionamento correspondente a cada uma das entradas do MCP23017 diretamente na porta do CI através de um fio tendo nível lógico de tensão em baixo. Para entradas configuradas com acionamento do tipo pulsado, foi verificado se o acionamento se deu em bordas de descida. Isso se deve ao fato de que as portas de entrada possuem um resistor de *pull-up*, tendo portanto acionamento em borda de descida para entradas pulsadas e acionamento em nível lógico alto ou baixo conforme configuração para o caso de entradas com acionamento do tipo estado. O resultado do teste manual pode ser visto no relatório do software na Figura 4.9, em que exhibe as entradas que foram acionadas conforme setando-se o nível lógico correspondente ao acionamento configurado na simulação, uma a uma. Um script possuindo uma instrução de aguardar acionamento/desativação de entrada também foi testado, justamente para avaliar se o

programa resume a execução das instruções de script após a entrada correspondente receber o evento neste cenário de aguardo de evento de entrada, como pode ser visto na Figura 3.16, na quarta instrução da imagem. Também houve êxito neste teste em questão, podendo várias instruções de aguardo de entrada serem encadeadas dentro de um mesmo script de execução. A funcionalidade mais avançada dentro do software simulador é o mecanismo de aguardo de evento de entrada, portanto, uma grande etapa dos testes se encerrou no momento em que este atuou corretamente.

Estando então finalizados todos os testes de controle da totalidade de saídas e entradas presente no módulo expansor (32 portas ao todo), o protótipo está pronto para atuar como um simulador junto à um RTAC de bancada dentro do laboratório de automação da COPEL, onde serão realizadas diversas simulações, e configuração de cenários diversos para que seja realizado os ajustes finos necessários para a garantia do funcionamento preciso do AUTOSIMOD. Por fim, após os desafios citados terem sido concluídos, o protótipo estaria pronto para desempenhar o comissionamento e ensaios do automatismo de novas subestações, substituindo a atual ferramenta analógica de simulação utilizada pela COPEL.

FIGURA 4.8 – MAPEAMENTO DAS 16 PORTAS PARA OS ENSAIOS E TESTES DO MC

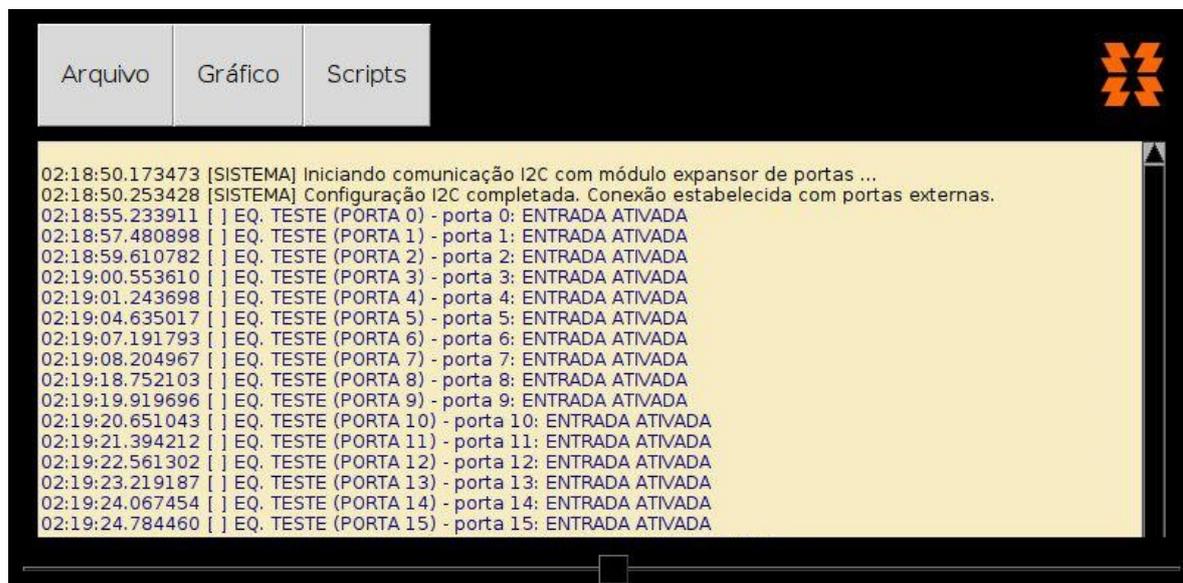


	Tipo	Equipamento	Porta de ativação	Porta de desativação	Tipo de acionamento	Acionamento em	Estad inicia
X	ENTRADA	DG1	1	0	PULSADO	BAIXO	DESATIVAD
X	SAÍDA	DG1	17	-	ESTADO	BAIXO	DESATIVAD
X	SAÍDA	DG1	16	-	ESTADO	BAIXO	ATIVADO
X	ENTRADA	DG2	3	2	PULSADO	BAIXO	DESATIVAD
X	SAÍDA	DG2	19	-	ESTADO	BAIXO	DESATIVAD
X	SAÍDA	DG2	18	-	ESTADO	BAIXO	ATIVADO
X	ENTRADA	DG3	5	4	PULSADO	BAIXO	DESATIVAD
X	SAÍDA	DG3	21	-	ESTADO	BAIXO	DESATIVAD

ENTRADAS: portas [0 - 15]
SAÍDAS: portas [16 - 31]

Fonte: Os Autores (2018)

FIGURA 4.9 – RELATÓRIO DE TESTE DO MC REALIZADO. AS 16 ENTRADAS DO MC ATIVAM CONFORME SINAIS RECEBIDOS DIRETAMENTE NA ENTRADA DO MCP23017



```

02:18:50.173473 [SISTEMA] Iniciando comunicação I2C com módulo expensor de portas ...
02:18:50.253428 [SISTEMA] Configuração I2C completada. Conexão estabelecida com portas externas.
02:18:55.233911 [ ] EQ. TESTE (PORTA 0) - porta 0: ENTRADA ATIVADA
02:18:57.480898 [ ] EQ. TESTE (PORTA 1) - porta 1: ENTRADA ATIVADA
02:18:59.610782 [ ] EQ. TESTE (PORTA 2) - porta 2: ENTRADA ATIVADA
02:19:00.553610 [ ] EQ. TESTE (PORTA 3) - porta 3: ENTRADA ATIVADA
02:19:01.243698 [ ] EQ. TESTE (PORTA 4) - porta 4: ENTRADA ATIVADA
02:19:04.635017 [ ] EQ. TESTE (PORTA 5) - porta 5: ENTRADA ATIVADA
02:19:07.191793 [ ] EQ. TESTE (PORTA 6) - porta 6: ENTRADA ATIVADA
02:19:08.204967 [ ] EQ. TESTE (PORTA 7) - porta 7: ENTRADA ATIVADA
02:19:18.752103 [ ] EQ. TESTE (PORTA 8) - porta 8: ENTRADA ATIVADA
02:19:19.919696 [ ] EQ. TESTE (PORTA 9) - porta 9: ENTRADA ATIVADA
02:19:20.651043 [ ] EQ. TESTE (PORTA 10) - porta 10: ENTRADA ATIVADA
02:19:21.394212 [ ] EQ. TESTE (PORTA 11) - porta 11: ENTRADA ATIVADA
02:19:22.561302 [ ] EQ. TESTE (PORTA 12) - porta 12: ENTRADA ATIVADA
02:19:23.219187 [ ] EQ. TESTE (PORTA 13) - porta 13: ENTRADA ATIVADA
02:19:24.067454 [ ] EQ. TESTE (PORTA 14) - porta 14: ENTRADA ATIVADA
02:19:24.784460 [ ] EQ. TESTE (PORTA 15) - porta 15: ENTRADA ATIVADA

```

Fonte: Os Autores (2018)

4.4 TESTE E RESULTADOS FINAIS

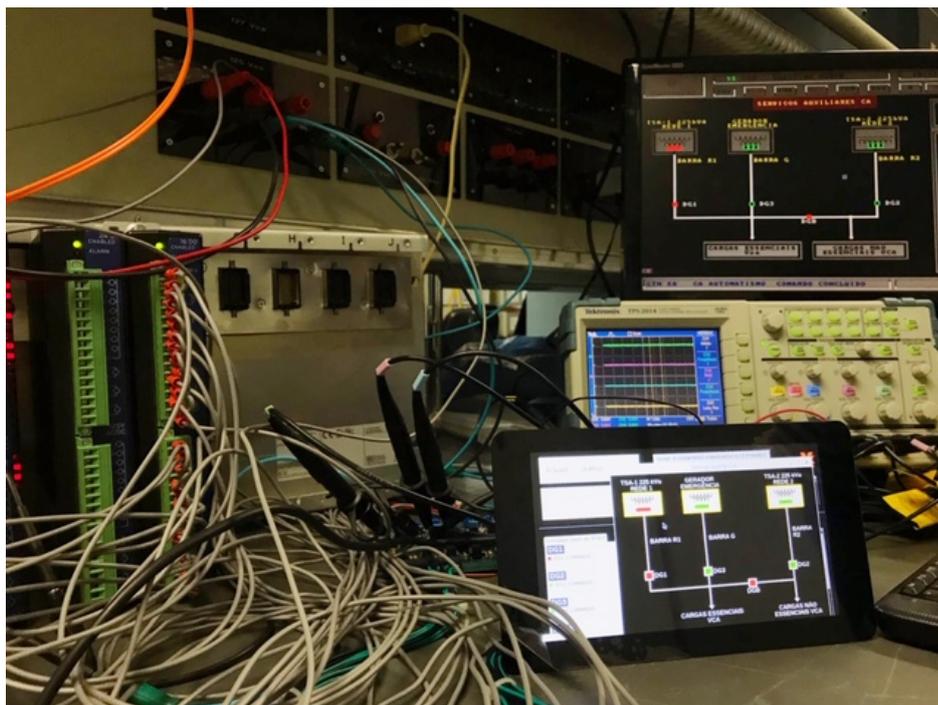
Dada a sequência de testes de integração entre o AUTOSIMOD (protótipo) e o controlador de automatismo em tempo real (RTAC), foram realizadas as últimas simulações para identificar possíveis falhas de implementação e redefinições de algumas variáveis de software que melhor se adequavam a necessidade apontada pelo setor de automação da Copel.

Na parte de software, houve alteração quanto a leitura dos dados emitidos pelo RTAC. Primeiramente haviam sido implementadas variáveis que verificavam, em tempo real, as portas lógicas de comunicação do protótipo de maneira linear, ou seja, era interpretado um sinal de cada vez nas portas de entrada do equipamento. Porém foi verificado, que sinais que eram transmitidos em um curto espaço de tempo ou simultaneamente apresentavam falha em sua leitura, pois somente um dos sinais era identificado na interface. Desta forma as variáveis de leitura dos sinais transmitidos pelo RTAC passaram a ser alocadas em THREADS, para que a identificação dos mesmos passassem a ser realizadas paralelamente, evitando-se assim possíveis falhas na recepção de dados transmitidos pelo controlador.

Ultrapassadas as modificações citadas, foi realizado um último teste de verificação entre o módulo desenvolvido e o RTAC. Neste teste, houve a validação por parte

dos engenheiros e técnicos do setor de automação da Copel e ficou demonstrado o sucesso no funcionamento do protótipo, ao verificar todos os dados transmitidos pelo RTAC de maneira eficiente. A Figura 4.10 mostra o teste realizado em bancada para verificar as lógicas de atuação do RTAC.

FIGURA 4.10 – TESTE FINAL DO FUNCIONAMENTO DO AUTOSIMOD SOB AS LÓGICAS DE ATUAÇÃO DO RTAC



Fonte: Os Autores (2019)

4.5 COMPARATIVO ENTRE SIMULADOR ANALÓGICO E O PROTÓTIPO DESENVOLVIDO

Conforme citado neste trabalho, uma das motivações para que se desenvolvesse um módulo simulador digital de serviços auxiliares, era a diminuição física deste dispositivo. Isto permitiria a mobilidade do equipamento para toda e qualquer subestação a ser testada, diferentemente do que acontecia anteriormente, quando o simulador analógico necessitava de um rack de telecomunicações para ser fixado. O seu transporte também era dificultado dado ao seu tamanho, o que praticamente inviabilizava a realização de testes além do local onde foi montado. Na Figura 4.10 é possível verificar o painel de botoeiras elétricas que tinha por objetivo simular os estados dos serviços auxiliares e toda a parte de cabeamento, que conectava as entradas e saídas do RTAC

ao simulador analógico. Além disso é pertinente destacar que este simulador é completamente dependente da ação de um usuário externo, tanto para os acionamentos das portas lógicas quanto para a interpretação e verificação das lógicas computacionais embarcadas no RTAC.

FIGURA 4.11 – ANTIGO MÓDULO SIMULADOR ANALÓGICO UTILIZADO NA COPEL



Fonte: Os Autores (2019)

O novo módulo simulador digital, mostrado na Figura 4.11, demonstra versatilidade tanto em tamanho quanto em funcionalidade, pois permite ser armazenado em qualquer caixa que possua volume mínimo de 9 litros (30x20x15 cm), o que facilita a sua mobilidade independente do local, e por possuir um software embarcado ao RPi com as trativas indicadas pelo RTAC, pode simular em tempo real 128 entradas e

saídas digitais de dispositivos atuantes nos serviços auxiliares de uma subestação.

Durante os testes de bancada do novo simulador, evidenciou-se a praticidade no modo de utilização do serviço prestado pelo mesmo e a facilidade de interação entre dispositivo e usuário, já que qualquer modificação das configurações de teste ou de criação de novos scripts de simulação de falhas, pode ser realizado através da tela TFT *touchscreen* associada ao protótipo, ou por teclado e mouse portáteis com saída USB que são facilmente conectados as portas I/O do computador RPi.

FIGURA 4.12 – NOVO MÓDULO SIMULADOR DIGITAL DO SISTEMA DE SERVIÇOS AUXILIARES DE UMA SUBESTAÇÃO UTILIZADO NA COPEL



Fonte: Os Autores (2019)

Desta forma é possível verificar o atingimento do objetivo deste trabalho, uma vez que as necessidades apontadas foram supridas e que a implementação do protótipo será imediata no setor de automação da Copel, de modo a aumentar a confiabilidade dos sistemas serviços auxiliares já em funcionamento em algumas subestações e de garantir os futuros projetos a serem desenvolvidos na área.

5 CONCLUSÃO

No transcorrer deste projeto foi possível verificar que a necessidade apontada pelo setor de automação da Companhia de Energia Elétrica do Paraná (Copel) era de fato notável, por se tratar da confiabilidade de sistemas que auxiliam no correto funcionamento de subestações que impactam diretamente na vida de milhares de pessoas.

O objetivo pelo qual este projeto foi iniciado tinha por característica, desenvolver e implementar um simulador digital que fosse capaz de aferir os mais variados cenários lógicos de atuação de um controlador de automatismo em tempo real (RTAC), de modo a identificar possíveis falhas que este possa apresentar por meio de suas lógicas embarcadas.

Em vista do cumprimento dos objetivos específicos deste trabalho, foi realizado o estudo dos SA da subestação CTN junto ao levantamento das necessidades do departamento de automação de SEs da COPEL, que revelaram a carência de um método modernizado de realização de ensaios do automatismo de novas subestações. A forma de simular e testar a lógica de automação embarcada no RTAC praticada atualmente pela COPEL, passa a ser tecnicamente defasada ao comparar-se com um sistema computadorizado e enxuto como o proposto por este trabalho. O modelo atualmente utilizado pela COPEL, faz uso de módulos analógicos de acionamento, operados manualmente através de botoeiras, o que torna este um processo desprovido de histórico, não havendo portanto um relatório da atividade das portas simuladas em tempo real, o que abre margem a erros na conferência do bom funcionamento da lógica de automação testada. Isso gera insegurança tanto para o departamento responsável pela automação, quanto para a própria concessionária de energia, que possui capital na casa de milhões em equipamentos presentes na planta de uma subestação com controle autônomo.

Diferente do que se tem utilizado até então pela COPEL, o emprego de um computador embarcado para controlar e registrar as portas de simulação, faz com que a confiabilidade do automatismo dos SA de uma SE aumente consideravelmente, visto que a interface de software permite aos engenheiros responsáveis a visualização em tempo real dos estados, acionamentos e relatório da simulação, através de uma tela digital *touchscreen*. Além disso, a praticidade na atuação das portas de estado

através de um toque na tela e a possibilidade de execução flexível de *scripts* de simulação, transformam o protótipo em uma ferramenta praticamente indispensável no comissionamento da automação dos SA em novas subestações.

Portanto, este trabalho buscou o desenvolvimento em hardware e software de um protótipo capaz de testar lógicas de automatismo de um controlador de sistema de serviços auxiliares (RTAC), que compõe todas as fontes e cargas de uma subestação. A implementação do protocolo I2C na conversação entre o computador embarcado e os módulos expansores de portas, permite implementar novas aplicações ao projeto, além de apenas a funcionalidade de testar lógicas de automatismo dos SA.

No teste de integração se obteve um resultado exitoso no controle das saídas do MC por execução de script via o ambiente de software. Neste teste foi evidenciado o correto funcionamento do protocolo I2C, protocolo este que intermedeia a comunicação do software com as referidas portas de saída, possibilitando, portanto, um importante passo para a execução dos testes finais, realizados com um RTAC de bancada nos ensaios laboratoriais de automação da COPEL. Além disso, a possibilidade de expansão no número de elementos controlados por protocolo I2C, traduziu-se em um aumento considerável no número de portas simuladas que podem ser adicionadas ao sistema fazendo uso do CI MCP 23017. Com isso, poderão ser ampliadas as funções de simulação do módulo para sistemas maiores dentro da SE, como os sistemas de proteção e controle por exemplo. Caso tais aplicações se tornem possíveis, o equipamento poderá servir como uma giga de testes, abrangendo se não toda, mas uma grande parte dos sistemas de uma SE.

As melhorias no comissionamento do automatismo de uma SE devido ao uso do protótipo descrito neste trabalho, se traduzem em uma adequação moderna às exigências e recomendações do órgão regulador ANEEL e também da ONS, no tocante à confiabilidade do sistema. Com uma automação imune a falhas, operações que seriam necessárias a realização na planta da SE, devido a problemas em lógicas de acionamento, ficam cada vez mais raras de acontecer, assim como acidentes e prejuízos materiais ainda mais inabituais, trazendo à SE um amadurecimento tecnológico e prático ainda maior. Logo, modernizar e aperfeiçoar a tecnologia de comissionamento de automação de uma SE, gera economia de operação à concessionária responsável, havendo motivos suficientes no estudo de prevenção de falhas através de simulação de ambientes e sistemas de produção ou gestão de energia elétrica, como é o caso

referenciado neste projeto.

Sendo assim foi possível perceber que o resultado foi atingido de maneira satisfatória, baseado nas projeções levantadas pelo corpo funcional do setor de automação da Copel e que o projeto dá margem a implementação de novos módulos a serem desenvolvidos em trabalhos futuros, na finalidade de se aumentar a amostragem de simulação em uma SE, não somente em serviços auxiliares mas em todos àqueles que compõem este sistema.

5.1 TRABALHOS FUTUROS

Propõem-se como trabalhos futuros, em um primeiro lugar, a execução de diversas simulações junto à um RTAC de bancada, avaliando o funcionamento conjunto dos módulos de estado e comandos da placa expansora de portas e corrigindo possíveis erros que prejudicam o devido funcionamento do protótipo.

O protótipo inicialmente tem por objetivo simular ambientes de funcionamento dos serviços auxiliares de uma subestação, mas não se limita somente à isso. Futuramente, devido a grande capacidade de processamento de dados oferecida pelo Raspberry Pi e possibilidades determinadas pelo software desenvolvido, será possível testar projetos vinculados a outras áreas da subestação, tais como, sistema de proteção, acionamento e automação.

O intuito final desse projeto é poder simular todas as variáveis apresentadas em uma subestação de energia. Essa proposta permitirá um estudo prévio de concepção de uma subestação, mitigando de forma precisa os possíveis erros de projeto e lógica do sistema de automação da mesma.

5.1.1 Transferência de Conhecimento e Protocolo de Implementação

Em virtude da necessidade de manutenções e/ou otimizações futuras que possam vir a ocorrer no protótipo ao qual este projeto se refere, foram estipuladas reuniões durante o período de testes em bancada junto ao corpo efetivo de trabalho da área de automação da Copel, para que haja o repasse de conhecimento aos engenheiros e técnicos do setor, assim como a criação de um manual dos recursos e de utilização do software embarcado no RPI.

Além disso, foi realizada uma consulta para averiguar a necessidade de confi-

gurar um protocolo de implementação do uso do protótipo nos ambientes de teste da Copel. No entanto, houve a negativa relacionada a esta questão, conforme relatado pela área responsável pelos testes e simulações na empresa. A justificativa dada é de que a aplicação do dispositivo desenvolvido se faz de maneira informal e, unicamente, voltada ao auxílio técnico e operacional dos funcionários envolvidos diretamente neste projeto.

REFERÊNCIAS

- AGERADORA. *O que é uma subestação?* 2016. Disponível em: <<https://www.ageradora.com.br/o-que-e-uma-subestacao/>>. Citado na página 18.
- ARDUINOECIA. Raspberry pi : Instale o raspbian e crie seu primeiro programa em python. <http://www.arduinoecia.com.br/2014/11/raspberry-pi-b-raspbian-python.html>, 2014. Citado na página 31.
- CARVALHO, E. M. *Protótipo de um dispositivo para captura de movimentos dos membros superiores em tempo real usando sensores inerciais*. Dissertação (Mestrado) — Universidade Federal de Uberlândia, 2016. Citado na página 29.
- FOUNDATION, R. P. Raspbian. <https://www.raspberrypi.org/downloads/raspbian/>, 2017. Citado na página 31.
- MUZY, G. L. C. O. *Subestações Elétricas*. Dissertação (Mestrado) — UFRJ, 2012. Citado na página 19.
- PATCHAVA, V.; KANDALA, H. B.; BABU, P. R. A smart home automation technique with raspberry pi using iot. In: IEEE. *Smart Sensors and Systems (IC-SSS), International Conference on*. [S.l.], 2015. p. 1–4. Citado 2 vezes nas páginas 29 e 30.
- SATO, K. C. *Transmissão de Potência em Corrente Contínua e Corrente Alternada: estudo comparativo*. Dissertação (Mestrado) — UNESP, 2013. Citado na página 19.
- SCHWEITZER. *SEL-2240 Axion*. [S.l.], 2018. Citado na página 25.
- SILVA, L. F. C. Windows 10 iot core insider preview e raspberry pi 3 - primeiras impressões. <https://www.embarcados.com.br/windows-10-iot-core-insider-preview-e-raspberry-pi-3/>, 2016. Citado na página 30.
- SUCKOW, C. F. *Subestações: Tipos, Equipamentos e Proteção*. Dissertação (Mestrado), 1999. Citado na página 19.