

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
TECNOLOGIA EM AUTOMAÇÃO INDUSTRIAL**

MARLON ROSA

**MONITORAMENTO DE INTERRUPÇÕES DE FORNECIMENTO EM
UMA REDE ELÉTRICA MONOFÁSICA RESIDENCIAL UTILIZANDO
MICROCONTROLADOR E SUPERVISÓRIO SCADABR**

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2016

MARLON ROSA

**MONITORAMENTO DE INTERRUPÇÕES DE FORNECIMENTO EM
UMA REDE ELÉTRICA MONOFÁSICA RESIDENCIAL UTILIZANDO
MICROCONTROLADOR E SUPERVISÓRIO SCADABR**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Automação Industrial, do Departamento Acadêmico de Eletrônica (DAELE), da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr.Sc. Abraham Elias Ortega Paredes

PONTA GROSSA

2016

Espaço destinado a elaboração da ficha catalográfica sob responsabilidade exclusiva do Departamento de Biblioteca da UTFPR.



TERMO DE APROVAÇÃO

(A SER FORNECIDA PELA SECRETARIA DO CURSO)

MONITORAMENTO DE INTERRUPÇÕES DE FORNECIMENTO EM UMA REDE ELÉTRICA MONOFÁSICA RESIDENCIAL UTILIZANDO UM MICROCONTROLADOR E SUPERVISÓRIO SCADABR

por

MARLON ROSA

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 29 de junho de 2016 como requisito parcial para a obtenção do título **Tecnólogo** em Automação Industrial. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Abraham Elias Ortega Paredes Dr.
Prof. Orientador

Jeferson José Gomes M.
Membro titular

Hugo Valadares Siqueira Dr.
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

Dedico este trabalho à minha esposa.

AGRADECIMENTOS

Agradeço a minha família pela paciência durante o desenvolvimento desse trabalho principalmente minha esposa pelo apoio e incentivo.

Agradeço a todos os professores do Departamento Acadêmico de Eletrônica que durante a jornada de graduação contribuíram repassando seus conhecimentos e experiências.

Agradeço ao orientador Prof. Dr. Abraham Elias Ortega Paredes por sua orientação na execução desse trabalho.

RESUMO

ROSA, Marlon. **Monitoramento de interrupções de fornecimento em uma rede elétrica monofásica residencial utilizando microcontrolador e supervisor ScadaBR..** 2016. 52f. Trabalho de Conclusão de Curso (Tecnologia em automação industrial) - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2016

O objetivo desse trabalho consiste em utilizar uma placa microcontrolada Arduino Uno com um sensor de tensão acoplado a uma de suas entradas analógicas, realizando o monitoramento da presença ou ausência de tensão em uma rede elétrica monofásica residencial. Esses dados coletados pelo sensor e placa Arduino Uno são transmitidos através da porta Universal Serial Bus (USB) utilizando o protocolo Modbus para o computador portátil com o *software* supervisor ScadaBr. Essas interrupções são registradas no banco de dados do *software* supervisor podendo gerar gráficos para confrontamento com os valores expressos na fatura de energia elétrica, valor esse denominado pela Agência Nacional de Energia Elétrica (ANEEL) como indicador Duração de Interrupção individual por unidade Consumidora (DIC).

Palavras-chave: Sistema supervisor. Modbus. Arduino Uno.

ABSTRACT

ROSA, Marlon. 2016. **Monitoring of power supply interruptions in a residential single-phase power supply using a microcontroller and supervisory software ScadaBR** 52f. Trabalho de Conclusão de Curso (Tecnologia em automação industrial) - Federal Technology University - Parana. Ponta Grossa, 2016.

The aim of this work is to use a microcontrolled board Arduino Uno with a voltage sensor coupled to one of its analog inputs, performing the monitoring of the presence or absence of voltage in a residential single-phase power grid. These data collected by the Arduino Uno board and sensor are transmitted via the Universal Serial Bus (USB) using the Modbus protocol to the laptop computer with the monitoring software ScadaBR. Such interruptions are recorded in the supervisory software database can generate charts for confrontation with the values expressed in the electricity bill, an amount called the National Electric Energy Agency (ANEEL) as an indicator Duration of individual Interruption per unit Consumer (DIC).

Keywords: Supervisory System. Modbus. Arduino Uno.

LISTA DE ILUSTRAÇÕES

Figura 1 : Arduino UNO	16
Figura 2: IDE Arduino	17
Figura 3: Exemplo de rede industrial	18
Figura 4: Conexão USB	21
Figura 5: Tipos de conectores USB.....	21
Figura 6: Rede Modbus.....	22
Figura 7: Comunicação mestre/escravo.....	23
Figura 8: Comunicação Modbus RTU	23
Figura 9: Endereços de memória dos registradores.....	25
Figura 10: Indicadores na fatura de energia.....	27
Figura 11:Consulta indicadores do conjunto	28
Figura 12: Indicadores na fatura	29
Figura 13: Sistema de supervisão da rede elétrica	30
Figura 14: Placa Arduino.....	31
Figura 15: Pacote Arduino.....	32
Figura 16: Driver dispositivo Arduino.....	32
Figura 17: Instalando Driver da placa Arduino	33
Figura 18: Configuração IDE.....	34
Figura 19: Ligação sensor de tensão	35
Figura 20: Conexão sensor de tensão	36
Figura 21: Primeira inicialização.....	37
Figura 22 (a) e (b): Configurando Data Source	38
Figura 23: Configurando parâmetros de comunicação.....	39
Figura 24: Pesquisando registradores.....	40
Figura 25: Variável criada	41
Figura 26: Watch list.....	41
Figura 27: Configuração detectores de eventos.....	42
Figura 28: Eventos na Watchlist.....	43
Figura 29: Gráfico ScadaBR	44
Figura 30: Gráfico dos desligamentos simulados.....	45
Figura 31: Eventos de interrupção	46
Figura 32: Simulação desligamento	46
Figura 33: Indicadores para cálculo	47

LISTA DE TABELAS

Tabela 1: Pinos USB	21
Tabela 2: Versões padrão USB.....	22
Tabela 3:Funções Modbus.....	24
Tabela 4: Custo dos recursos.....	30
Tabela 5: Testes de leitura.....	45
Tabela 6: Apuração valor DIC mensal	48

LISTA DE ABREVIATURAS E SIGLAS

A/D	Analógico/Digital
ANEEL	Agência Nacional de Energia Elétrica
ASCII	<i>American Standard Code for Information Interchange</i>
CLP	Controlador Lógico Programável
CRC	<i>Cyclical Redundancy Check</i>
DIC	Duração de Interrupção Individual por Unidade Consumidora
DMIC	Duração da Maior Interrupção contínua por unidade Consumidora
FIC	Frequência de Interrupção por unidade Consumidora
IDE	<i>Integrated Development Environment</i>
LED	<i>Light Emitting Diode</i>
PLC	<i>Programmable Logic Controller</i>
PWM	<i>Pulse Width Modulation</i>
RTU	<i>Remote Terminal Unit</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SO	Sistema Operacional
USB	<i>Universal Serial Bus</i>
UF	Unidade Federativa
VCA	Volts em Corrente Alternada
VCC	Volts em Corrente Contínua

SUMÁRIO

1 INTRODUÇÃO	13
1.1 TEMA DA PESQUISA	14
1.2 DELIMITAÇÃO DO TEMA.....	14
1.3 PROBLEMA	14
1.4 HIPÓTESE	14
1.5 OBJETIVOS.....	14
1.5.1 Objetivo Geral	14
1.5.2 Objetivos Específicos	15
1.6 JUSTIFICATIVA	15
1.7 MÉTODO DA PESQUISA	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 PLATAFORMA ARDUINO.....	16
2.1.1 Arduino Uno	16
2.1.2 Integrated development environment (IDE) Arduino	17
2.2 SISTEMA SUPERVISÓRIO	18
2.2.2 Componentes lógicos de um sistema SCADA	19
2.3 MODOS DE COMUNICAÇÃO.....	19
2.3.1 Comunicação por consulta (<i>polling</i>)	19
2.3.2 Comunicação por interrupção (<i>report by exception</i>).....	20
2.4 COMUNICAÇÃO USB.....	20
2.5 PROTOCOLO MODBUS.....	22
2.5.1 Modelo mestre/escravo	22
2.5.2 Mensagens Modbus RTU.....	23
2.5.2.1 Códigos das funções.....	24
2.5.2.2 Endereçamento lógico dos dados	24
2.5.2.3 Checagem de erros	25
3 DURAÇÃO DE INTERRUÇÃO INDIVIDUAL POR UNIDADE CONSUMIDORA OU PONTO DE CONEXÃO (DIC)	27
3.1 CÁLCULO DE COMPENSAÇÃO POR VIOLAÇÃO INDICADOR DIC.....	28
4 PROCEDIMENTOS DE EXECUÇÃO	30
4.1 INSTALAÇÃO PLACA ARDUINO UNO.....	30
4.1.1 INSTALAÇÃO SENSOR DE TENSÃO	34
4.2 INSTALANDO E CONFIGURANDO O SOFTWARE SCADABR	36
5 RESULTADOS	44
6 CONCLUSÃO	49
REFERÊNCIAS	50

1 INTRODUÇÃO

Em meados dos anos de 1970 e 1980 com avanço da eletrônica e surgimento dos microprocessadores. Os microcomputadores passaram a ser parte integrante das indústrias. Nesse momento surgiram então os primeiros sistemas supervisórios com a função de monitorar processos industriais (MORAES & CASTRUCCI, 2013)

Os sistemas supervisórios permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou instalação física. Tais informações são coletadas através de equipamentos de aquisição de dados e, em seguida, manipulados, analisados, armazenados e, posteriormente, apresentados ao usuário (SILVA & SALVADOR, 2005).

Com essa evolução tecnológica foi possível a criação dos microcontroladores, que diferem dos processadores por já possuírem os periféricos, como memórias, conversor Analógico Digital (A/D) e demais integrados. Esses têm como diferencial sua versatilidade para inúmeras aplicações em diversos segmentos como, indústria, eletrodomésticos e outros.

Esses recursos tecnológicos estão sendo aplicados até nos lares modernos. Sistemas de automação e supervisão em residências tem se popularizado e o microcontrolador serve como ferramenta muito eficaz em diversas aplicações. Podemos citar, exemplos de controle de iluminação, sistema de segurança, controle de sistema de irrigação, além de muitos eletrodomésticos que são microcontrolados.

Contudo há um produto essencial para que isso funcione, a energia elétrica. Quando seu fornecimento não é de qualidade pode gerar transtornos e até prejuízos financeiros.

Para isso no Brasil existe uma agência que regula o setor de energia elétrica. A Agência Nacional de energia elétrica (ANEEL), tem como missão:

A Agência Nacional de Energia Elétrica conduz seu trabalho com base em princípios e valores afinados com a expectativa da sociedade brasileira, de uma atuação equilibrada, transparente e comprometida com o interesse público (ANEEL, 2016).

Como forma de controlar a qualidade da prestação de serviço das empresas fornecedoras, a ANEEL estabeleceu indicadores de qualidade que devem ser cumpridos pelas empresas e são apresentados nas faturas de energia. Contudo seria viável termos nossos recursos para o monitoramento da qualidade da energia.

1.1 TEMA DA PESQUISA

Monitoramento de indicador de qualidade individual em uma rede elétrica.

1.2 DELIMITAÇÃO DO TEMA

Monitoramento por sistema supervisório (ScadaBR) de interrupções de fornecimento de energia em uma rede elétrica monofásica residencial utilizando um Arduino UNO. A plataforma Arduino tem seu projeto bem desenvolvido possibilitando recursos vastos em termos de materiais e informações para realização do monitoramento. Também há possibilidade de expansão de variáveis monitoradas se necessário, já que a mesma possui 6 entradas analógicas.

1.3 PROBLEMA

O problema trata do acompanhamento e monitorização das interrupções de fornecimento em uma rede elétrica residencial para possíveis queixas quanto aos tempos de interrupção, haja vista que estas proporcionam multas a distribuidora de energia em forma de desconto na fatura em favor do cliente. No contexto atual as interrupções de energia são contabilizadas pela companhia fornecedora e não existe qualquer acompanhamento pelo cliente.

1.4 HIPÓTESE

Realizar o monitoramento das interrupções de energia utilizando o microcontrolador ATmega 328P que integra a placa Arduino Uno e reportar os dados coletados ao *software* supervisório ScadaBR, que por sua vez armazenará as informações. Fazer uso desses dados para determinar o valor do indicador de qualidade individual atrelado aos tempos de interrupção, entendê-lo e quantificá-lo. Entender os critérios e forma de cálculo para ressarcimento por violação desse indicador.

1.5 OBJETIVOS

1.5.1 Objetivo Geral

Fornecer os valores de tensão da rede elétrica e apresentá-los no *software* supervisório inclusive gerando alarmes, gráficos dos valores provendo informações das interrupções ocorridas na rede e demonstrar os critérios e fórmula de cálculo do indicador DIC como também o método de cálculo dos ressarcimentos ao cliente fazendo uso dos dados coletados.

1.5.2 Objetivos Específicos

Para execução do monitoramento das interrupções na rede elétrica os seguintes passos serão necessários:

- Utilizar software ScadaBR num computador portátil para monitoramento do fornecimento de energia.
- Utilizar sensor de tensão na rede elétrica e conectá-lo ao microcontrolador Atmega328R da plataforma do Arduino.
- Configurar parâmetros de comunicação entre o *software* ScadaBR e a plataforma Arduino.
- Configurar a plataforma do Arduino para reportar variáveis de níveis de energia ao supervisor.
- Realizar configuração do sistema supervisor e as variáveis a serem monitoradas.

1.6 JUSTIFICATIVA

Durante evento de interrupção de fornecimento de energia elétrica na residência do autor desse trabalho suscitaram-se algumas dúvidas quanto ao tempo total, início e fim da interrupção, a correção das informações prestadas na fatura, como também a interpretação das informações por parte do autor como cliente.

O monitoramento se mostra interessante pois não se tem certeza dos dados fornecidos na fatura, pois a única referência são as informações fornecidas pela empresa.

Os recursos como software e hardware necessários para implementação de supervisão da unidade consumidora são de fácil aquisição e baixo valor, o software supervisor não incidirá qualquer custo na aplicação pois se trata de projeto *Open Source* com os recursos necessários.

1.7 MÉTODO DA PESQUISA

A pesquisa em relação ao objetivo será descritiva, detalhando os procedimentos para integração do equipamento ao sistema supervisor e sua viabilidade. Quanto ao objeto de estudo será de campo no equipamento instalado na unidade e a coleta de dados se dará pelo sistema aquisitando os dados da rede elétrica e armazenando no banco de dados do *software*, com possibilidade de geração de gráficos pelo usuário.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 PLATAFORMA ARDUINO

O projeto Arduino teve seu início em 2005 pelo Prof. Massimo Banzi e o pesquisador suíço David Cuartielles na cidade de Ivrea na Itália e surgiu como uma ideia de fazer os estudantes trabalhar em tecnologia e baixo custo.

Além de seu custo baixo outro atrativo é sua flexibilidade para desenvolvimento dos variados tipos de projetos. Estas características de preço e flexibilidade fizeram com que mais de 500.000 placas Arduino fossem vendidas pelo mundo até 2015, além das placas clone (MCROBERTS 2015, p. 24).

2.1.1 Arduino Uno

Existem várias versões de placas Arduino, uma das mais populares e uma ótima opção multiuso é a placa Arduino Uno, baseada no microcontrolador ATmega328. Possui 14 pinos de entradas/saídas digitais (no qual 6 podem ser usadas como saídas *Pulse Width Modulation* (PWM), 6 entradas analógicas, conexão USB, cabo de força, botão de reset. (EVANS;NOBLE;HOCHENBAUM, 2013). A Figura 1 mostra uma placa Arduino Uno.

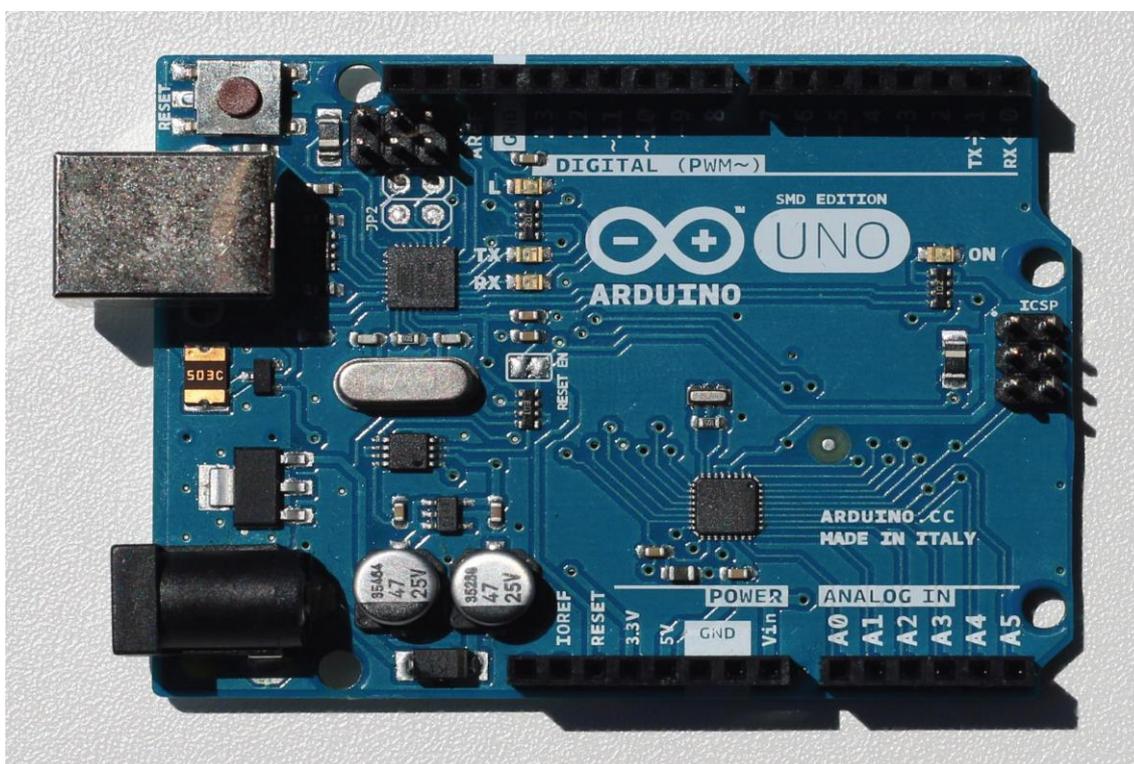


Figura 1 : Arduino UNO
Fonte: Autoria: Lanbridge (2015)

2.1.2 Integrated development environment (IDE) Arduino

No site do projeto Arduino (www.arduino.cc) é possível obter o software de desenvolvimento para programação da placa, estão à disposição as versões para instalação nos sistemas operacionais (SO) Windows, Mac e Linux. Segundo McRoberts (2011) a IDE é dividida em três partes, *Toolbar* no topo, *Sketch Window* ou código no centro e janela de mensagens na base. O *software* sendo executado é exemplificado na Figura 2 com a barra de ferramentas destacada em vermelho, em azul a área para o código no centro e em verde logo abaixo a janela de mensagens.

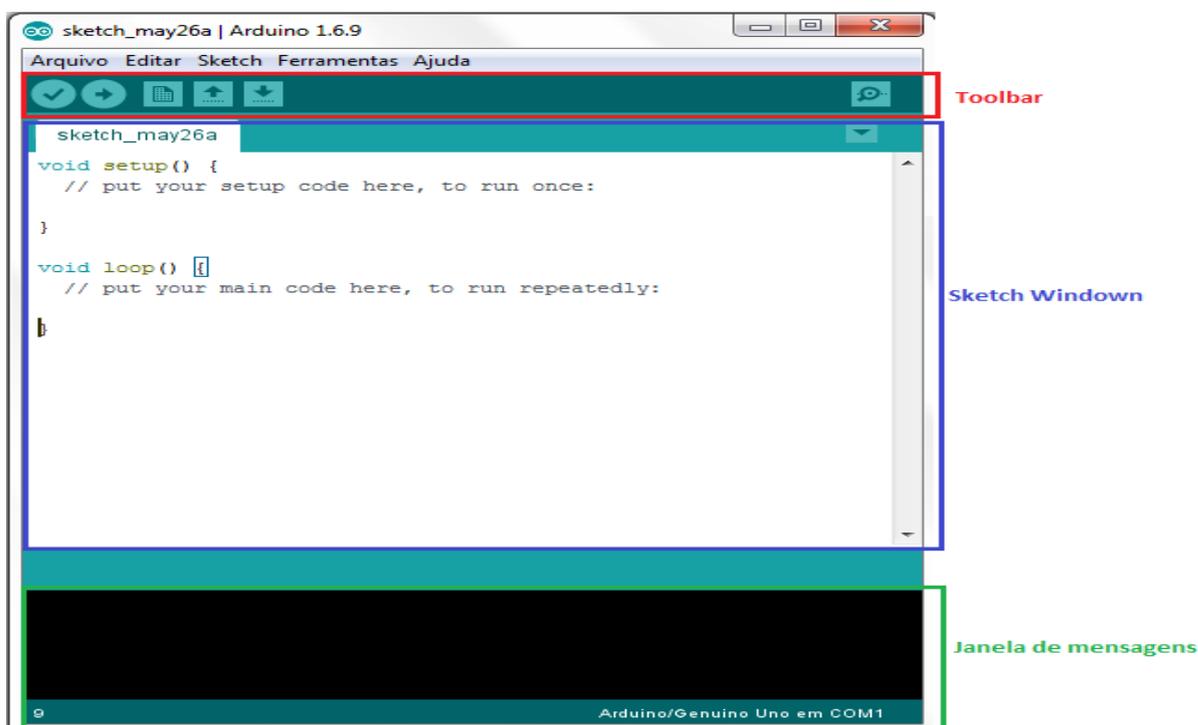


Figura 2: IDE Arduino
Fonte: Autoria própria (2016)

No topo há a barra de menus onde está localizada a *Toolbar* com as funções visualizadas da esquerda para a direita:

- Verificar; verifica o código escrito.
- Carregar; carrega o código na placa Arduino.
- Novo; cria novo programa.
- Abrir; abre programa salvo anteriormente.
- Salvar; salva o código.

Para McRoberts (2011) essas são as funções mais utilizadas.

2.2 SISTEMA SUPERVISÓRIO

O sistema supervisório tem como funções monitorar e controlar processos ou grandezas dos mais variados tipos, e segundo Albuquerque e Alexandria (2009, p. 213) é responsável pela coleta e transferência de informações lógicas e analógicas sobre o estado corrente do sistema, pela exibição desses dados na sala de controle.

Segundo Coelho (2009, p. 4) o objetivo principal do sistema SCADA é propiciar uma interface de alto nível do operador com o processo informando-o em tempo real de todos os eventos de importância da planta.

Na Figura 3 é apresentado um exemplo de rede industrial em que no nível mais baixo estão os sensores responsáveis por ler as variáveis físicas do sistema, como por exemplo, vazão, pressão e transmiti-las ao Controlador Lógico Programável (CLP). Também no mesmo nível aplicados em campo estão os atuadores que atuam no sistema, sendo comandados pelo CLP, alguns exemplos de atuadores são válvulas, solenoides, etc. O controlador reporta ao supervisório informações recebidas do sensores e atuações realizadas na planta. O sistema SCADA processa as informações recebidas da planta e envia para ao sistema gerencial.

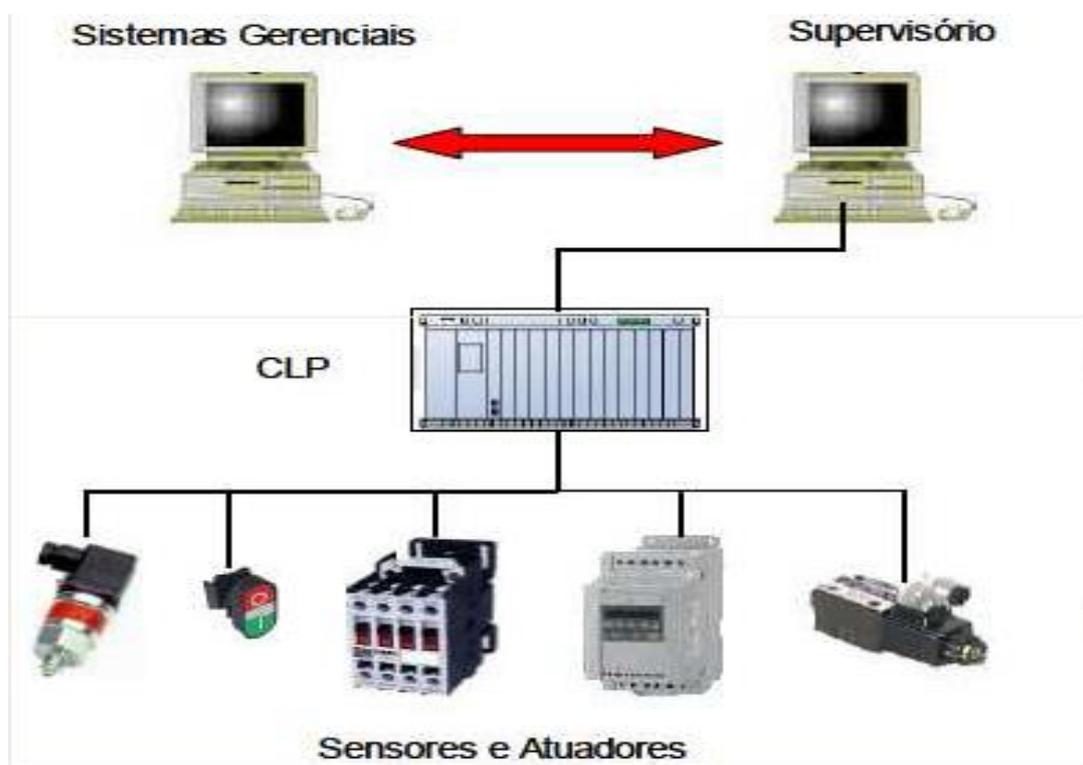


Figura 3: Exemplo de rede industrial
Fonte: Casillo (2008)

2.2.2 Componentes lógicos de um sistema SCADA

Para Silva e Salvador (2005, p. 3) as principais funções dos sistemas SCADA são divididas em blocos e funções, que basicamente são:

- Núcleo de processamento;
- Comunicação com *Programmable Logic Controller* (PLC) / *Remote Terminal Unit* (RTU)
 - Gerenciamento de Alarmes.
 - Históricos e Banco de Dados.
 - Lógicas de programação interna (Scripts) ou controle.
 - Interface gráfica.
 - Relatórios.
 - Comunicação com outras estações SCADA.
 - Comunicação com Sistemas Externos / Corporativos.

2.3 MODOS DE COMUNICAÇÃO

Um sistema SCADA tem como tarefa essencial trocar informações entre equipamentos (PLCs e RTUs) e outros sistemas SCADA ou ainda sistema de gestão. Essas informações são transmitidas através de um protocolo pré-definido que poder ser aberto ou de acesso restrito e segundo Moraes e Castrucci (2013, p. 121) pode ser por consulta (*polling*) ou por interrupção (*report by exception*).

2.3.1 Comunicação por consulta (*polling*)

Neste modo de comunicação o dispositivo que tem o comando total sobre uma hierarquia inferior também conhecido como mestre é responsável por toda comunicação e a faz de forma sequencial com os escravos, que são chamados assim por apenas executarem qualquer ação somente após serem solicitados pelo mestre, nesse caso retornando as informações solicitadas. Nessa situação segundo Moraes e Castrucci (2013, p. 122) caso uma estação não responda durante um período de tempo pré-determinado as solicitações, novas tentativas de comunicação serão realizadas antes de se declarar *time-out* e se passe a próxima estação.

Para efeitos de decisão do modo de comunicação as seguintes vantagens e desvantagens devem ser levadas em conta.

Vantagens da comunicação por *polling*:

- Simplicidade no processo de aquisição de dados.
- Inexistência de colisões no tráfego da rede.
- Permite mensurar a largura de banda de comunicação.

Desvantagens:

- Incapacidade por parte das estações remotas, de comunicar situações que tenham necessidade imediata.
- Quanto maior o número de estações maior o tempo de espera.

2.3.2 Comunicação por interrupção (*report by exception*)

Nesse modo de comunicação a estação remota quando há alteração significativa nos valores pré-definidos inicia o processo de comunicação com a estação central. O sistema é implementado de forma a detectar os erros e colisões.

Conforme Moraes e Castrucci (2013, p. 122) antes de iniciar a transmissão à estação remota verifica se o meio está ocupado. Caso esteja, aguarda um tempo pré-determinado para tentar iniciar e assim. Tal procedimento é repetido sucessivamente até encontrar o meio de comunicação disponível para transmissão.

A estação central realiza varreduras por *polling* para verificar as estações que eventualmente não estejam comunicando por se encontrarem em estado de falha.

Vantagens da comunicação por interrupção:

- Evita transferência de informação desnecessária, diminuindo o tráfego da rede.
- Rápida detecção de informação urgente.

Desvantagens:

- A estação central consegue detectar falhas na ligação a partir de um tempo pré-determinado apenas quando efetuar um *polling* no sistema.
- É necessária ação do operador para obter valores atualizados.

2.4 COMUNICAÇÃO USB

Em meados de 1990 empresas do mercado de computadores criaram o padrão *Universal Serial Bus* (USB). Este foi implementado para facilitar a conexão de periféricos aos computadores pessoais permitindo a conexão de até 127 dispositivos em uma porta USB segundo Torres (2001, p. 344). A Figura 4 mostra a flexibilidade de apenas uma porta USB do microcomputador a que está conectada um teclado, que por sua vez estão conectados um mouse, joystick e um hub. O hub que tem como função multiplicar as portas disponíveis e conectar ao microcomputador através de sua conexão com o teclado, os dispositivos *scanner*, impressora e câmera digital.

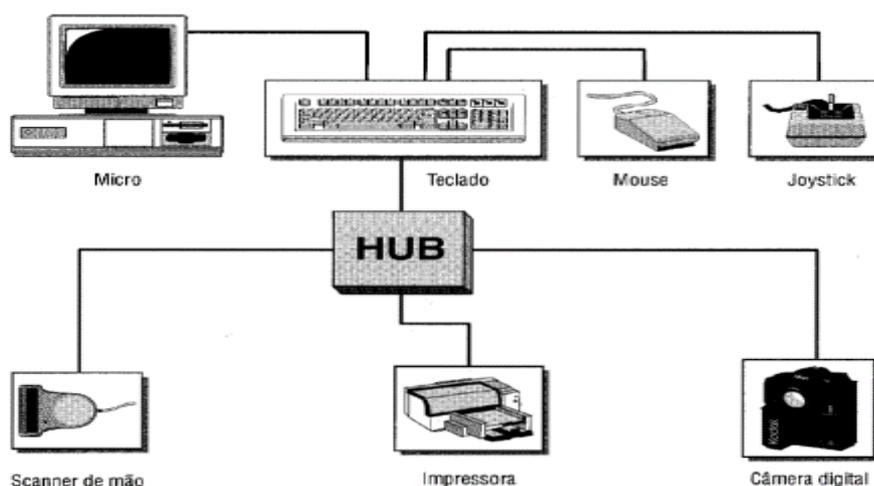


Figura 4: Conexão USB
Fonte: Torres (2011)

Com o padrão USB os usuários se beneficiam da facilidade da tecnologia *plug and play*, ou seja, basta conectar e usar não necessitando realizar configurações de *hardware*, para Zelenovsky (2002, p. 691) a tecnologia apresenta como grande vantagem a conexão de periféricos sem abrir o computador. Os conectores são padronizados em conectores tipo A e B conforme a Figura 5.

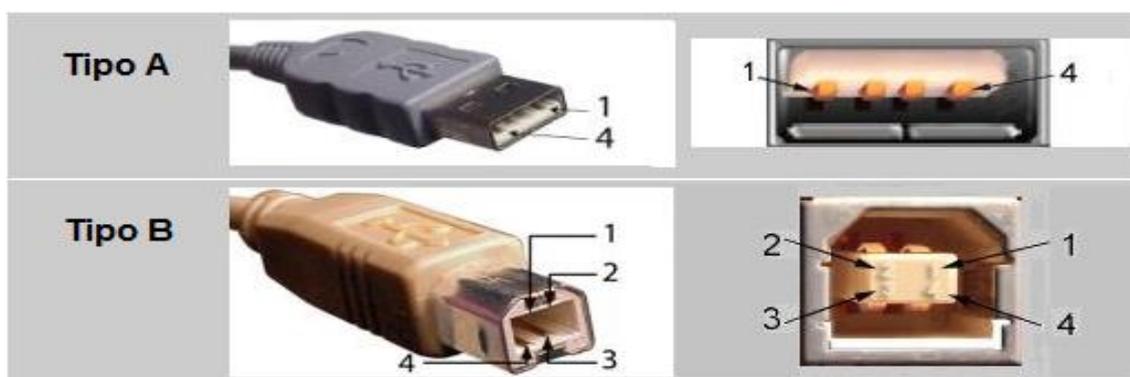


Figura 5: Tipos de conectores USB
Fonte: Adaptado de Novaetrônica (2016)

O cabo USB tem sua limitação de comprimento máximo de 5 metros para perfeito funcionamento sem perda de dados. A Tabela 1 apresenta a descrição dos pinos e respectivas funções.

Pino	Sinal	Cor	Descrição
1	VCC	Vermelho	+5V
2	D-	Branco	Data -
3	D+	Verde	Data +
4	GND	Preto	Ground

Fonte: Autoria própria (2016)

Desde o lançamento do padrão USB suas versões foram sendo atualizadas e com sua evolução foi elevada as taxas de transferência de dados, na Tabela 2 são apresentadas algumas versões com suas respectivas taxas de transferência.

Versão	Ano	Velocidade
1.0	1996	1,5 Mbps
2.0	2000	480 Mbps
3.0	2008	4.8 Gbps
3.1	2013	10 Gbps

Fonte: Autoria própria (2016)

2.5 PROTOCOLO MODBUS

O protocolo *modbus* foi desenvolvido no ano de 1979 pela empresa Modicom com intuito de estabelecer a comunicação entre dispositivos mestre/escravo. Segundo Albuquerque e Alexandria (2009, p. 118) “nesse modelo os escravos não podem dialogar entre si, toda comunicação deve passar por um mestre”. A Figura 6 apresenta uma conexão entre um CLP mestre conectado a três escravos Modbus, os números apresentados representam seus endereços na rede.

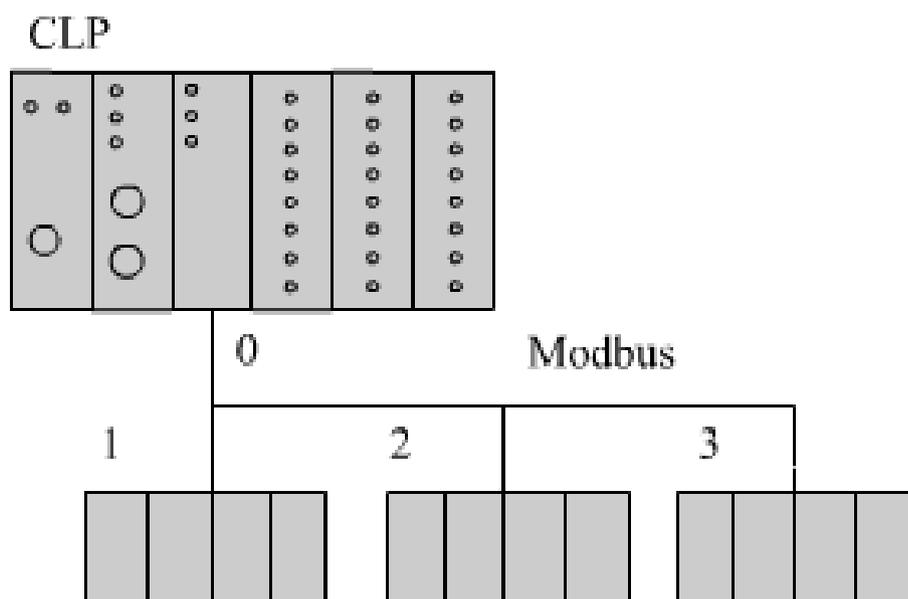


Figura 6: Rede Modbus
Fonte: Real Time Automation (2016)

2.5.1 Modelo mestre/escravo

Cada mestre pode ter até 247 escravos, e a comunicação é sempre iniciada pelo mestre que envia uma mensagem que pode ser de dois tipos:

-*Unicast*: o mestre envia uma mensagem de requisição para um escravo e o mesmo responde ao mestre.

Na Figura 7 é demonstrado a troca de informações entre mestre e escravo. O mestre envia a solicitação, o escravo executa e retorna a resposta. O mestre recebe a informação.

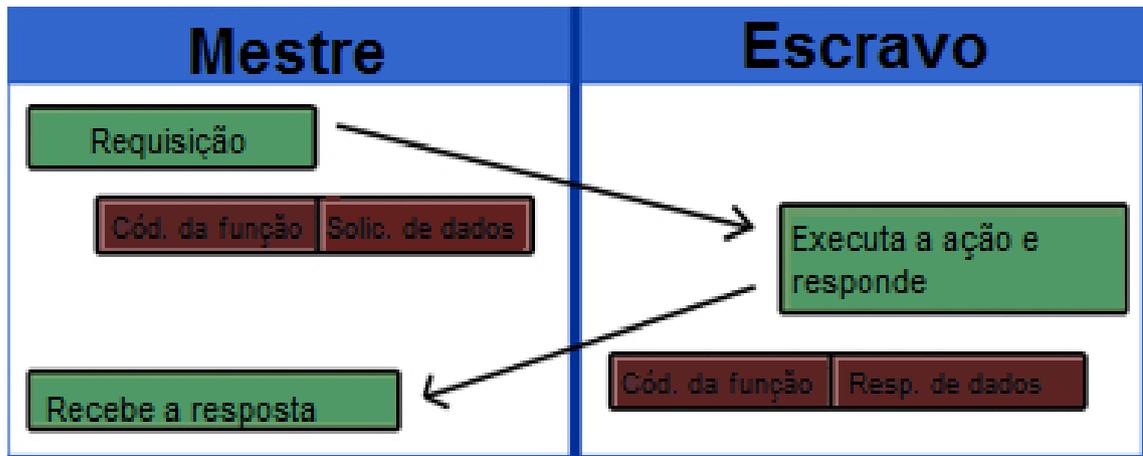


Figura 7: Comunicação mestre/escravo
Fonte: Adaptado de Real Time Automation (2016)

-*Broadcast*: O mestre envia uma mensagem a todos os escravos, porém não há retorno de resposta para o mestre.

Segundo Moraes e Castrucci (2007, p. 171) existem duas formas seriais para o protocolo Modbus quanto ao modo de transmissão: RTU e *American Standard Code for Information Interchang* (ASCII).

2.5.2 Mensagens Modbus RTU

As mensagens em Modbus são formadas por *frames* ou quadros que podem ser de consulta ou de resposta. A Figura 8 mostra a transação entre um dispositivo mestre e um escravo e o formato das informações trocadas entre mestre e escravo.

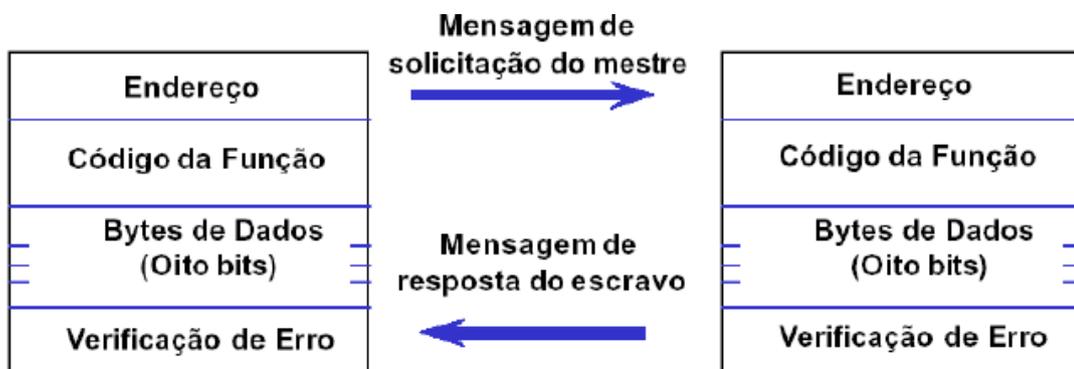


Figura 8: Comunicação Modbus RTU
Fonte: Laboratório de garagem (2016)

2.5.2.1 Códigos das funções

Os tipos dos dados de 1 *bit* nas trocas de mensagens no protocolo modbus podem ser:

- Bobinas (*coils*): podem ser lidas ou escritas no escravo
- Entradas (*inputs*): podem ser apenas lidas do escravo.

Os dados de 16 *bits* (*registers*) podem ser dos tipos:

- Retentivos (*holding*): podem ser lidos ou escritos no escravo.
- Entradas: somente podem ser lidos do escravo.

Quando o mestre transmite uma mensagem para um escravo, embute na mensagem o código do serviço solicitado (Albuquerque e Alexandria, 2009, p. 120), a Tabela 3 mostra as funções disponíveis.

Tabela 3: Funções Modbus

Código em hexadecimal	Função	Descrição
1	<i>Read coil status</i>	Leitura de saída digitais
2	<i>Read input status</i>	Leitura de entradas digitais
3	<i>Read holding register</i>	Leitura de registros
4	<i>Read input registers</i>	Leitura de registros de entradas
5	<i>Force single coil</i>	Escrita de uma saída digital
6	<i>Preset single register</i>	Escrita de um único registro
F	<i>Force multiple coils</i>	Escrita de várias saídas digitais
10	<i>Preset multiple registers</i>	Escrita de vários registros

Fonte: Adaptado de Albuquerque e Alexandria (2009)

2.5.2.2 Endereçamento lógico dos dados

Cada tipo de dados, sejam eles de um único *bit* (*coils* ou *inputs*) ou de 16 *bits* (*holdings* ou *inputs*) podem ter até 9999 variáveis. São determinadas faixas de endereços lógicos para cada tipo de dado. Segundo Albuquerque e Alexandria (2009, p. 123) os endereços são:

- *Coils*: 00001 a 09999 (saída discreta).
- *Inputs*: 10001 a 19999 (entrada discreta).
- *Inputs registers*: 30001 a 39999 (entradas analógicas).
- *Holding registers*: 40001 a 49999 (saídas analógicas).

Na Figura 9 podemos observar na primeira coluna os endereços na memória do escravo de acordo com o tipo das variáveis descritos na segunda coluna. Na

terceira coluna é apresentado o código de acordo com o tipo da variável e o que se pretende, ler ou escrever os dados, nos tipos entradas analógicas e digitais não são apresentados os códigos para escrita pois essas só podem ser lidas.

Endereços dos Registradores	Memória Dispositivo	Comandos	
		Leitura	Escrita
0XXXX	Solenóides Saídas Discretas	1	5,15
1XXXX	Entradas Digitais	2	N/D
3XXXX	Entradas Analógicas	4	N/D
4XXXX	Registradores de Memória	3	6,16

Figura 9: Endereços de memória dos registradores
Fonte: Coelho (2014)

2.5.2.3 Checagem de erros

No protocolo modbus RTU é utilizada para checagem de erros o método *Cyclical Redundancy Check* (CRC) 16. O método CRC 16 é constituído de 2 bytes (ALBUQUERQUE; ALEXANDRIA, 2009).

Para Tanenbaum (2009, p. 160) o CRC trata-se de um código polinomial e se baseia no tratamento de strings de bits como representação de polinômios com coeficientes 1 e 0.

O CRC 16 tem como características principais:

- Detectar grande quantidade de erros de transmissão, isolados ou em rajadas.

- Possuir algoritmo de cálculo mais complexo.
- Pode ser calculado por hardware ou software.

Segundo Seixas Filho (2012, p. 5) os passos na execução do CRC 16.

- Cada bit da mensagem (m) codificada em binário é considerado como coeficiente de um polinômio $M(X)$ de base 2.

- A mensagem é deslocada para a esquerda de r posições, onde r é o número de bits do CRC (ordem do polinômio verificador = número de bits da representação do polinômio verificador – 1).

- A mensagem deslocada é dividida por um polinômio característico $G(X)$.

- $T(X)$ é transmitida.
- O receptor divide $T(X)$ por $G(X)$.
- Se o resultado for 0, existe grande probabilidade de a mensagem estar correta, caso contrário existe um erro.

3 DURAÇÃO DE INTERRUÇÃO INDIVIDUAL POR UNIDADE CONSUMIDORA OU PONTO DE CONEXÃO (DIC)

O indicador de interrupção individual por unidade consumidora ou ponto de conexão trata de interrupções maiores que 3 minutos e cujo valor de tensão for menor que 70% do valor nominal de fornecimento. É expresso em horas e centésimos de hora e obrigatoriamente discriminada na fatura de energia elétrica. Para Companhia elétrica Light (2015) o DIC refere-se à quantidade de horas que uma unidade consumidora permaneceu sem energia. Segundo Agência Nacional de Energia Elétrica (2016) "A continuidade do fornecimento é avaliada pela ANEEL através de subdivisões das distribuidoras, denominadas Conjuntos Elétricos" Na Figura 10 é apresentado o campo da fatura com os indicadores individuais de qualidade com suas metas e valores apurados. Este campo mostra o mês a que se refere os dados das metas, como também o conjunto a que a unidade consumidora está associada.

INDICADORES DE QUALIDADE				
Mês de referência: Julho/2010				
Conjunto: Centro				
Indicadores	Apurado Mensal	Meta Mensal	Meta Trimestral	Meta Anual
DIC	9,57	10,30	20,60	41,20
FIC	3,00	7,70	15,50	31,00
DMIC	6,85	5,50	-	-
DIC - Duração de interrupção individual FIC - Frequência de interrupção individual DMIC - Duração máxima de interrupção contínua				
VALOR DO ENCARGO DE USO DO SISTEMA DE DISTRIBUIÇÃO:				
RS 8,60				
<small>O cliente tem o direito de solicitar a qualquer tempo a apuração dos indicadores DIC, FIC e DMIC e também receber uma compensação, caso sejam violadas as metas de continuidade individuais – mensal, trimestral e anual – relativos à unidade consumidora de sua responsabilidade.</small>				

Figura 10: Indicadores na fatura de energia

Fonte: Companhia Light (2015)

No site oficial da ANEEL é possível fazer a consulta as metas de indicadores bastando a indicação da Unidade Federativa (UF), município, ano e conjunto para se ter acesso as metas definidas para a distribuidora de energia elétrica. Na Figura 11 é apresentada a tela para consulta das metas de indicadores no site (<http://www2.aneel.gov.br/aplicacoes/srd/indqual/default.cfm>).

Divulgação dos limites dos indicadores DIC, FIC, DMIC e DICRI

A continuidade do fornecimento é avaliada através de indicadores que mensuram a frequência e a duração das interrupções ocorridas nos consumidores. Ressalta-se que, similarmente a outros indicadores no mundo, os indicadores são apurados para as interrupções maiores que 3 minutos, sendo admitidos alguns expurgos na sua apuração. Os indicadores de continuidade são os seguintes:

- Duração equivalente de interrupção por unidade consumidora (DEC): Intervalo de tempo que, em média, no período de apuração, em cada unidade consumidora do conjunto considerado ocorreu descontinuidade da distribuição de energia elétrica.
- Frequência equivalente de interrupção por unidade consumidora (FEC): Número de interrupções ocorridas, em média, no período de apuração, em cada unidade consumidora do conjunto considerado.
- Duração de interrupção individual por unidade consumidora (DIC): Intervalo de tempo que, no período de apuração, em cada unidade consumidora ou ponto de conexão ocorreu descontinuidade da distribuição de energia elétrica.
- Frequência de interrupção individual por unidade consumidora (FIC): Número de interrupções ocorridas, no período de apuração, em cada unidade consumidora ou ponto de conexão.
- Duração máxima de interrupção contínua por unidade consumidora ou ponto de conexão (DMIC): Tempo máximo de interrupção contínua de energia elétrica, em uma unidade consumidora ou ponto de conexão.
- Duração da interrupção individual ocorrida em dia crítico por unidade consumidora ou ponto de conexão (DICRI): Corresponde à duração de cada interrupção ocorrida em dia crítico, para cada unidade consumidora ou ponto de conexão.

A continuidade do fornecimento é avaliada pela ANEEL através de subdivisões das distribuidoras, denominadas Conjuntos Elétricos. Existem limites para indicadores associados a cada conjunto, que podem ser consultados a seguir. Ressalta-se que o conjunto elétrico pode ter abrangência variada. Conjuntos grandes podem abranger mais de um município, ao mesmo tempo que alguns municípios podem possuir mais de um conjunto. Os limites dos indicadores DIC e FIC são definidos para períodos mensais, trimestrais e anuais. O limite do indicador DMIC é definido para períodos mensais. O limite do indicador DICRI é definido para cada interrupção em dia crítico. O assunto está regulamentado no Módulo 8 dos Procedimentos de Distribuição - PRODIST. As informações referentes aos indicadores de continuidade estão disponíveis na fatura de energia elétrica. Informações adicionais devem ser obtidas com a distribuidora.

PR - PARANÁ Seleção um Município Informe o ano Seleção um Conjunto Elétrico

Fechar

Figura 11: Consulta indicadores do conjunto
Fonte: ANEEL (2016)

3.1 CÁLCULO DE COMPENSAÇÃO POR VIOLAÇÃO INDICADOR DIC.

Segundo ANEEL (2016) o ressarcimento deve proceder da seguinte forma:

"No caso de violação do limite de continuidade individual dos indicadores DIC, FIC e DMIC em relação ao período de apuração (mensal, trimestral ou anual), a distribuidora deverá calcular a compensação ao consumidor acessante do sistema de distribuição, inclusive àqueles conectados em DIT, e efetuar o crédito na fatura, apresentada em até dois meses após o período de apuração (ANEEL, 2016)".

Para cálculo da indenização do consumidor pela violação do indicador DIC a seguinte forma deve ser aplicada segundo ANEEL.

$$Valor = \left(\frac{DICv}{DICp} - 1 \right) DICp \times \frac{EUSDmédio}{730} \times kei$$

Onde temos:

- DICv; Duração do somatório das interrupções expresso em horas e centésimos de hora verificado no período.
- DICp; limite estabelecido para o período em horas e centésimos de hora que podemos consultar na fatura ou no site da ANEEL.
- EUSDmédio; média aritmética dos encargos de uso do sistema de distribuição, o valor do EUSD é do mês corrente é fornecido na fatura de energia.

- 730; número médio de horas do mês.
- kei; coeficiente de majoração que pode ser 15 para baixa tensão (igual ou menor que 1000 Volts), 20 para consumidores de média tensão (maior que 1 KV e menor ou igual a 69 KV) e Alta Tensão(maior que 69 KV e menor que 230 KV), valor da tensão contratada é apresentado junto ao campo com os indicadores de qualidade da fatura.

Na Figura 12 são apresentados os valores necessários para cálculo do ressarcimento por violação do DIC, circulado em amarelo o EUSD, em vermelho as metas mensal, trimestral e anual e destacado em verde o valor do DIC apontado pela distribuidora de energia. Com o valor da tensão contratado tem-se o valor de kei=15 por se tratar de baixa tensão.

Indicadores de Qualidade					FS [1.7.61.0]
Conjunto: SABARA			Mes 03/2016		Tensao Contratada:
	DIC	FIC	DMIC	EUSD (R\$)	127 volts
Realizado Mensal:	0,00 h	0,00	0,00 h	26,66	Limite faixa adequada de Tensao:
Limite Mensal:	4,83 h	3,17	2,69 h		117 - 133 volts
Limite Trimestral:	9,67 h	6,35			
Limite Anual:	19,34 h	12,70			

Figura 12: Indicadores na fatura
Fonte: Autoria própria (2016)

4 PROCEDIMENTOS DE EXECUÇÃO

Para execução do projeto os recursos necessários foram adquiridos. A Tabela 4 apresenta os recursos obtidos e seus devidos custos.

Tabela 4: Custo dos recursos

Quantidade (UN)	Descrição	Valor (R\$)
1	Software ScadaBR	0,00
1	Placa Arduino UNO	45,00
1	Sensor de tensão	12,00
1	Outros	17,30
Total		74,30

Fonte: Autoria própria (2016)

Para o perfeito funcionamento do *software* supervisorio operando com a placa Arduino Uno e sensor de tensão, conjunto esse responsável pelo monitoramento da rede elétrica, foram necessárias algumas etapas entre instalação dos *softwares* supervisorio e IDE Arduino como também a instalação da placa microcontrolada e sensor. Na Figura 13 é mostrado de forma simplificada em blocos os componentes, suas funções e resultados esperados.

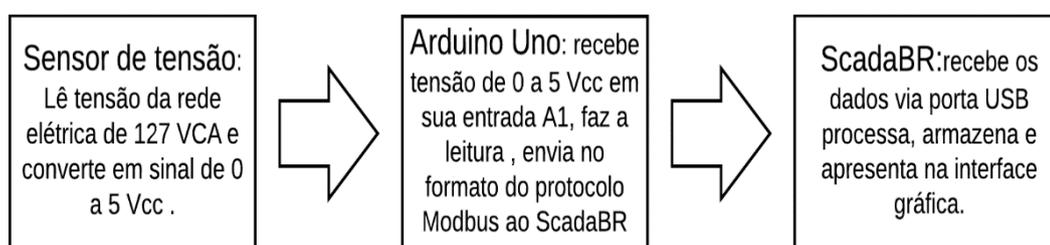


Figura 13: Sistema de supervisão da rede elétrica
Fonte: Autoria própria (2016)

4.1 INSTALAÇÃO PLACA ARDUINO UNO

Para leitura da tensão da rede foi escolhido o microcontrolador ATmega328 na plataforma Arduino UNO, por atender as necessidades técnicas e ser o mais acessível em termos financeiros da linha Arduino. Esse receberá o sinal do sensor conectado à rede elétrica, em valores indicados pelo fabricante. O microcontrolador tem a função de receber esse sinal analógico converte-lo em digital, processá-lo e nesse caso, utilizando meio de comunicação enviar, esses dados ao *software* ScadaBR.

A placa Arduino Uno adquirida foi fornecida com cabo USB para conexão desse hardware ao computador portátil, a plataforma de desenvolvimento pode ser alimentada com fonte externa ou pela própria conexão USB que fornece 5 Volts para alimentação do periférico, a Figura 14 apresenta o equipamento e cabo.

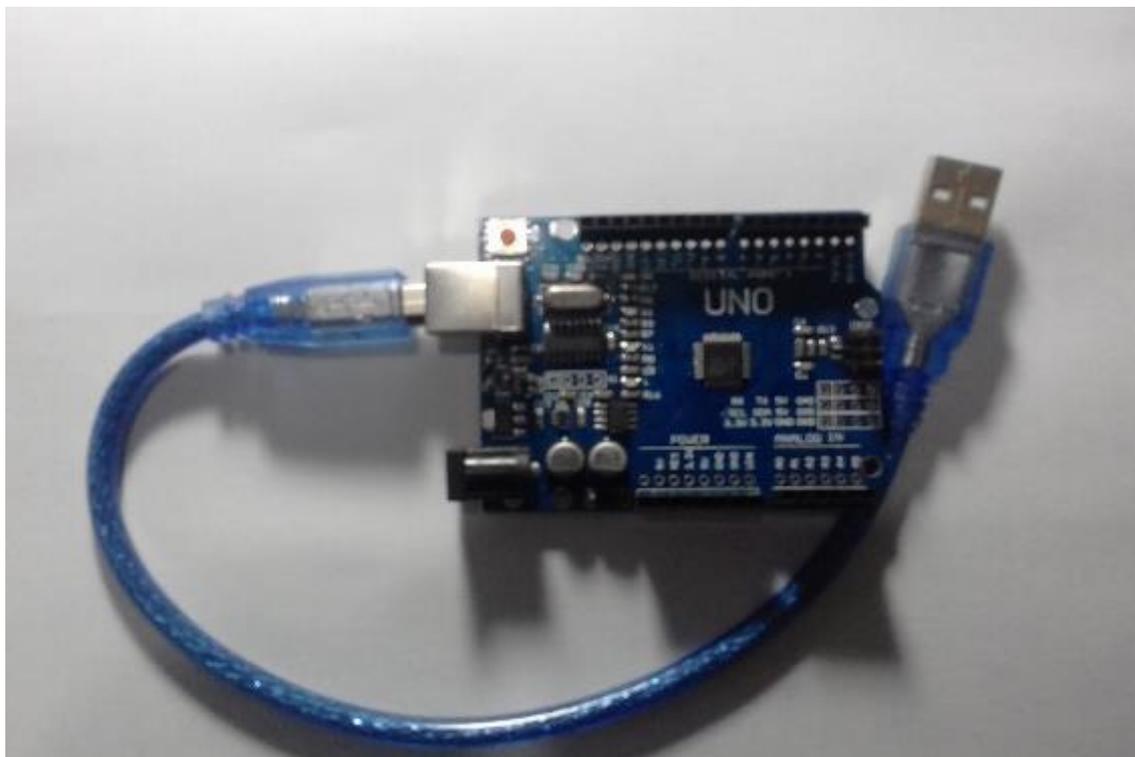


Figura 14: Placa Arduino
Fonte: Aatoria Própria (2016)

Para o funcionamento do microcontrolador como desejado existe a necessidade de programar as instruções ou código. E para isso os desenvolvedores do projeto fornecem um ambiente de desenvolvimento chamado de IDE. No site oficial do projeto Arduino foi realizado o *download* do *software* para programação da placa microcontrolada, versão para o sistema operacional Windows, sistema este instalado no computador portátil. Esses arquivos estão disponíveis no site do projeto Arduino e basta abrir o arquivo executável de nome Arduino como é apresentado na Figura 15.

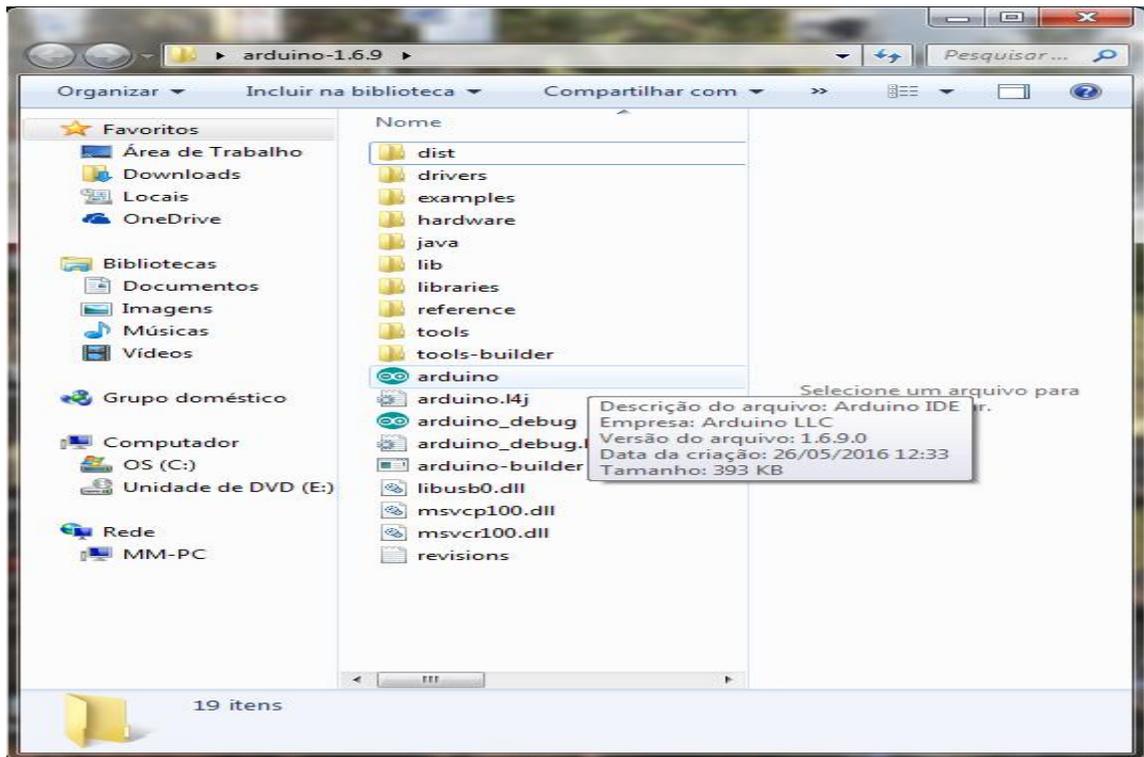


Figura 15: Pacote Arduino
Fonte: Autoria Própria (2016)

O equipamento foi conectado a uma porta USB do computador portátil e o *Lighting Emitting Diode* (LED) de indicação *ON* disponível no Arduino Uno indicou que a placa estava ligada. Como os dispositivos USB são *plug and play* o SO indicou a instalação de um novo hardware, porém o *driver* para o funcionamento com o *software* não foi encontrado necessitando a instalação manual conforme mostra a Figura 16.

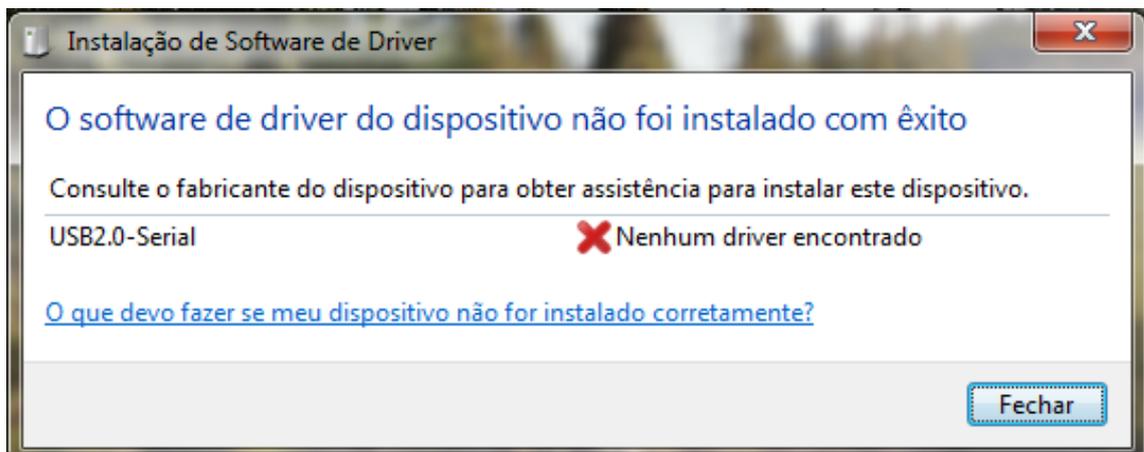


Figura 16: Driver dispositivo Arduino
Fonte: Autoria Própria (2016)

Para correção do problema foi necessário instalar o *driver* que é disponibilizado no pacote descarregado, juntamente com o *software* para programação do Arduino. Para isso foi necessário acessar o menu do SO painel de controle/gerenciador de dispositivos e atualizar o *driver* indicando o caminho até o arquivo que está disponível no diretório *drivers*, dentro do arquivo descarregado do site do desenvolvedor. A Figura 17 apresenta a execução do gerenciador de dispositivos para instalação do *driver* do dispositivo.

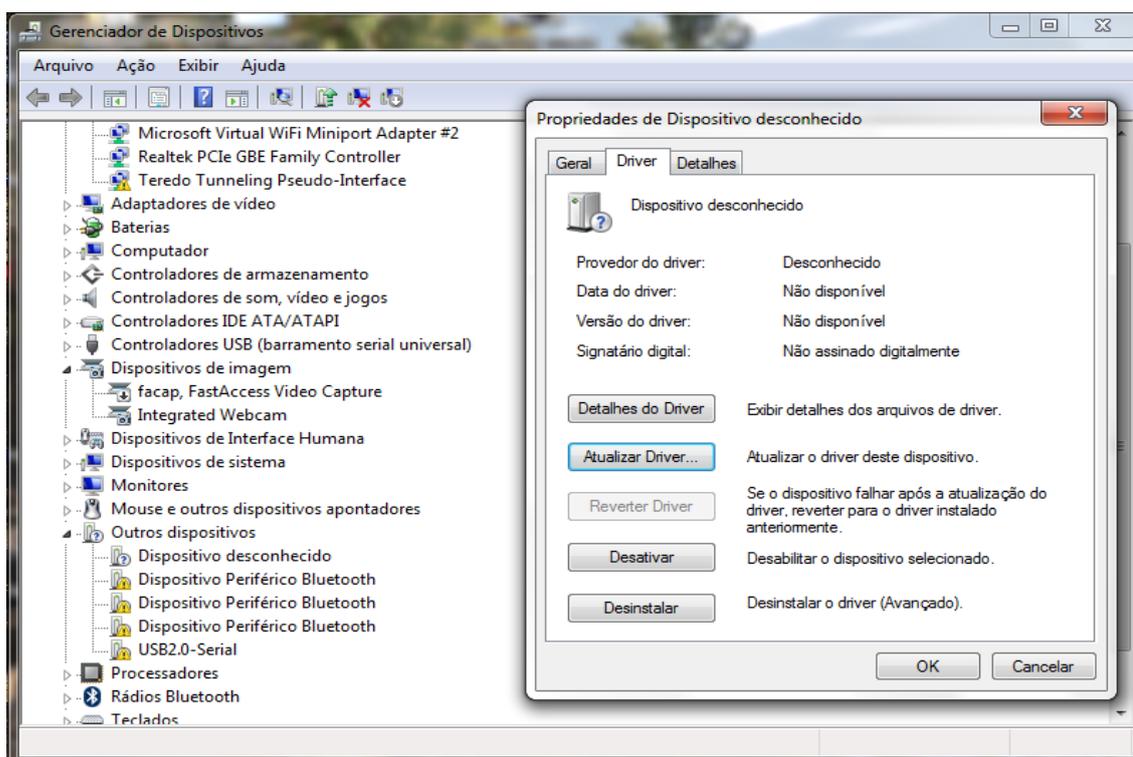


Figura 17: Instalando Driver da placa Arduino
Fonte: Autoria própria (2016)

Após reconhecimento da placa Arduino pelo SO foi procedida a execução da IDE. Itens básicos são necessários para possibilitar a conexão entre a placa e o software do fabricante: configuração do modelo de placa, e a porta serial criada na conexão do dispositivo USB ao computador. Na aba ferramentas da IDE há as opções placa e logo abaixo a opção porta, essas foram configuradas como Arduino Uno e porta serial COM6 como pode ser visualizado na Figura 18.

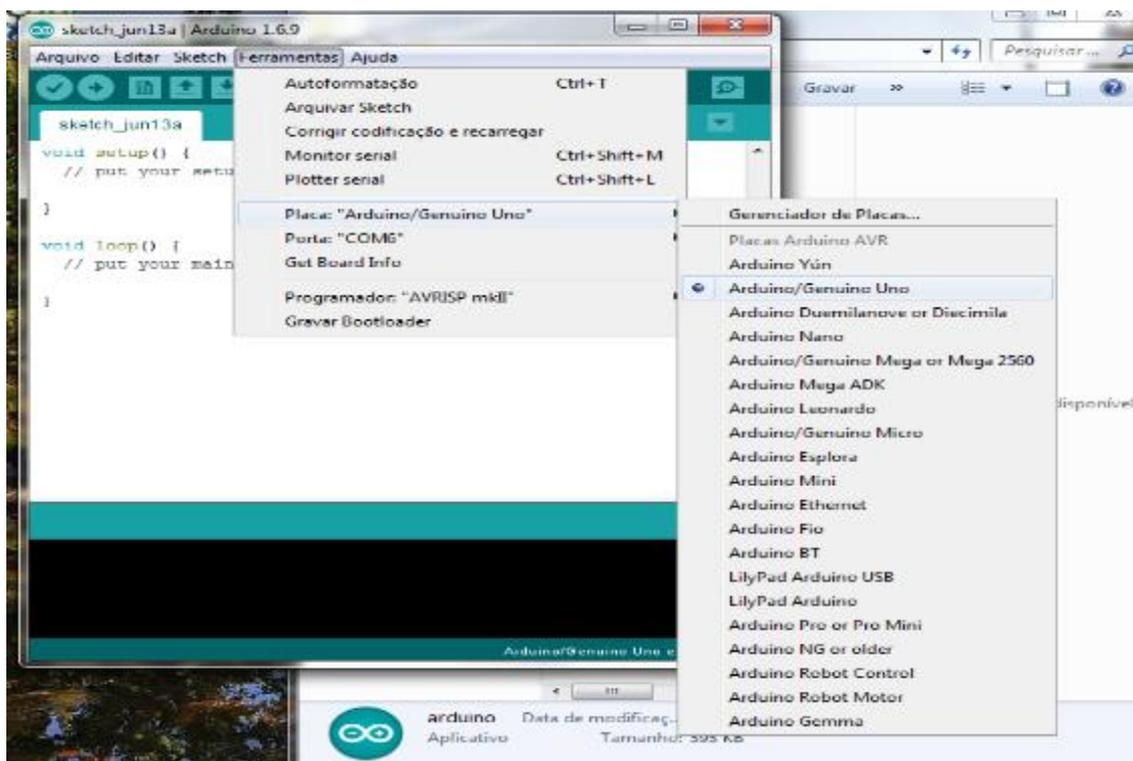


Figura 18: Configuração IDE
Fonte: Autoria própria (2016)

Para verificar o correto funcionamento da plataforma Arduino foi utilizado código disponibilizado pelo desenvolvedor que exemplifica algumas programações possíveis para a plataforma que estão disponíveis na aba Arquivo/Exemplos/. Foi escolhido aleatoriamente visto que a intenção era apenas observar se a carga do programa seria executada. Feita a verificação do código esse apresentou na janela de mensagem compilação terminada. Então foi utilizada na *Toolbar* o ícone enviar, sendo procedido o carregamento na placa Arduino do código demonstrando o perfeito funcionamento do equipamento sendo possível prosseguir com as demais etapas até o funcionamento do sistema completo.

4.1.1 Instalação Sensor de Tensão

Para sensoriamento da tensão da rede elétrica faz-se necessário o condicionamento do sinal elétrico pois a tensão da rede objeto de monitorização é 127 Volts enquanto a porta analógica do microcontrolador suporta 5 Volts. Esse sensor recebe o sinal da rede em 127 Volts e apresenta proporcionalmente um valor entre 0 e 5 Volts em sua saída que está conectada a entrada analógica A1 do microcontrolador. O sensor utilizado foi fornecido pela empresa MASTERWALKER

SHOP e veio acompanhado de tutorial para instalação na Figura 19 é apresentado em exemplo de ligação do sensor fornecido pela empresa.

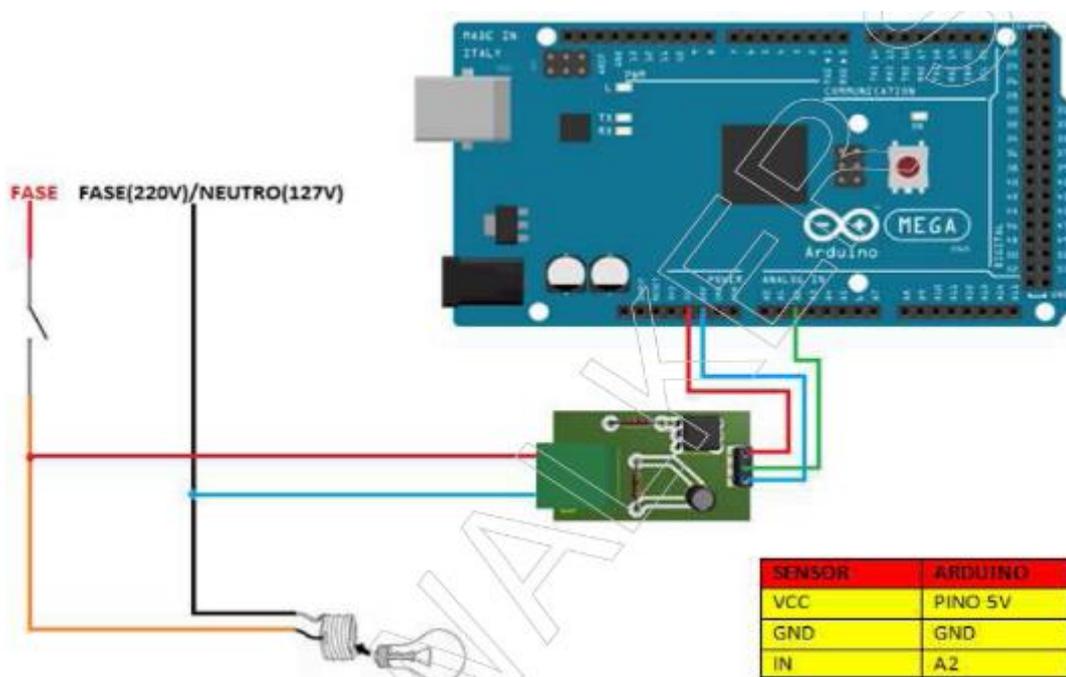


Figura 19: Ligação sensor de tensão
Fonte: MASTERWALKER SHOP (2016)

Foi usado cabo com conector para conexão a uma tomada residencial. Entre a alimentação que chega pelo cabo ao sensor foi inserido uma chave interruptora para simular desligamentos. Na Figura 20 é apresentada foto do sensor com as conexões indicadas e todo o conjunto instalado restando apenas a instalação do software supervisor para funcionamento do sistema.

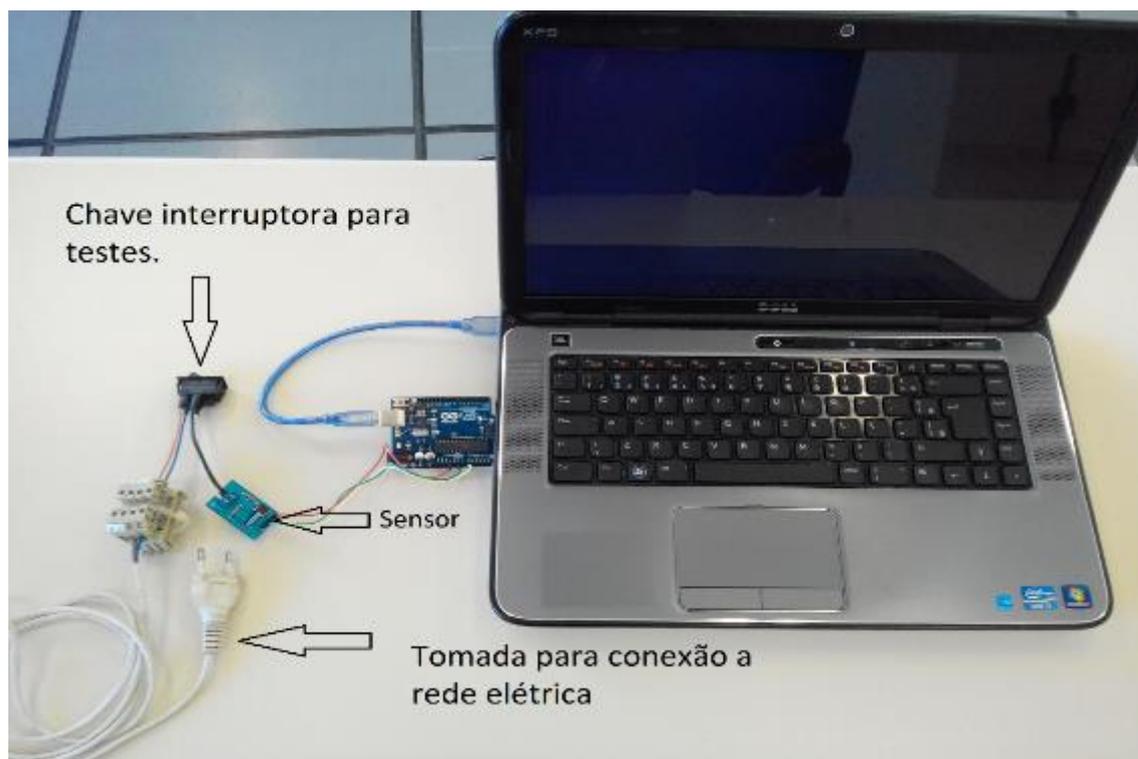


Figura 20: Conexão sensor de tensão
Fonte: Autoria própria (2016)

4.2 INSTALANDO E CONFIGURANDO O SOFTWARE SCADABR

Para armazenar os dados das interrupções da rede elétrica e possibilitar a visualização dessas informações de forma gráfica foi necessário um *software* supervisor. Existe várias opções no mercado e podemos citar algumas como Elipse Scada, Simatic WinCC (Siemens) e tantos outros. A opção pelo ScadaBR se deu devido ao caráter do seu projeto *Open Source*, ou seja, sem qualquer custo diferentemente dos programas anteriormente citados.

Para instalação do *software* supervisor ScadaBR foi necessária descarregar o arquivo disponível no site. Após realização do download no site do desenvolvedor do *software* da versão 1.0CE, versão para o sistema operacional Windows instalado no computador portátil que supervisionaria a rede elétrica. Foi executado arquivo de instalação conforme passo do tutorial disponível no site. A primeira execução do *software* foi satisfatória, após a instalação foi apresentada a tela inicial de *login*. O usuário e senha iniciais são, admin e admin. Conforme demonstra a Figura 21.

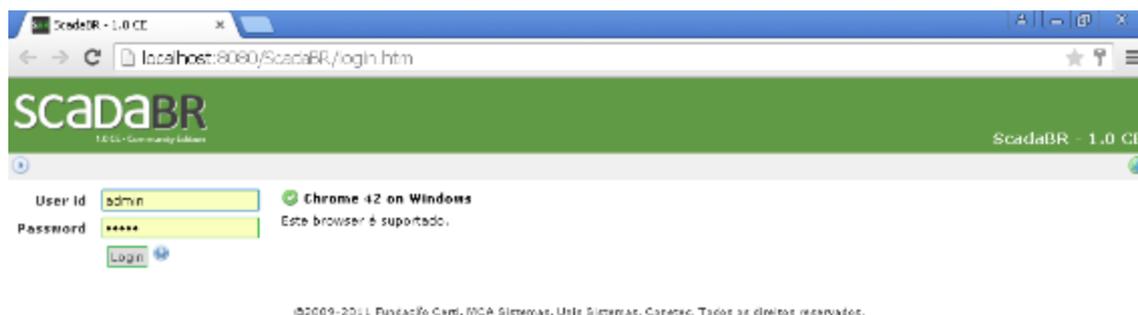


Figura 21: Primeira inicialização

Fonte: Autoria própria (2016)

Para se estabelecer a comunicação e troca de informações é necessário criar no banco de dados do supervisório uma configuração com parâmetros de comunicação, protocolo e variáveis a serem aquiridas do microcontrolador. Essa configuração de parâmetros no supervisório ScadaBR que tornará possível a interação e troca de informações adequadamente entre o hardware Arduino UNO e programa.

O primeiro passo para configuração foi criar uma *Data Source* com o tipo de protocolo e meio de transmissão de dados, a opção correspondente foi Modbus Serial. Essa *Data Source* para fácil identificação foi nominada como Rede_residencial. As Figuras 22 (a) e (b) mostram a criação dessa *Data Source*.

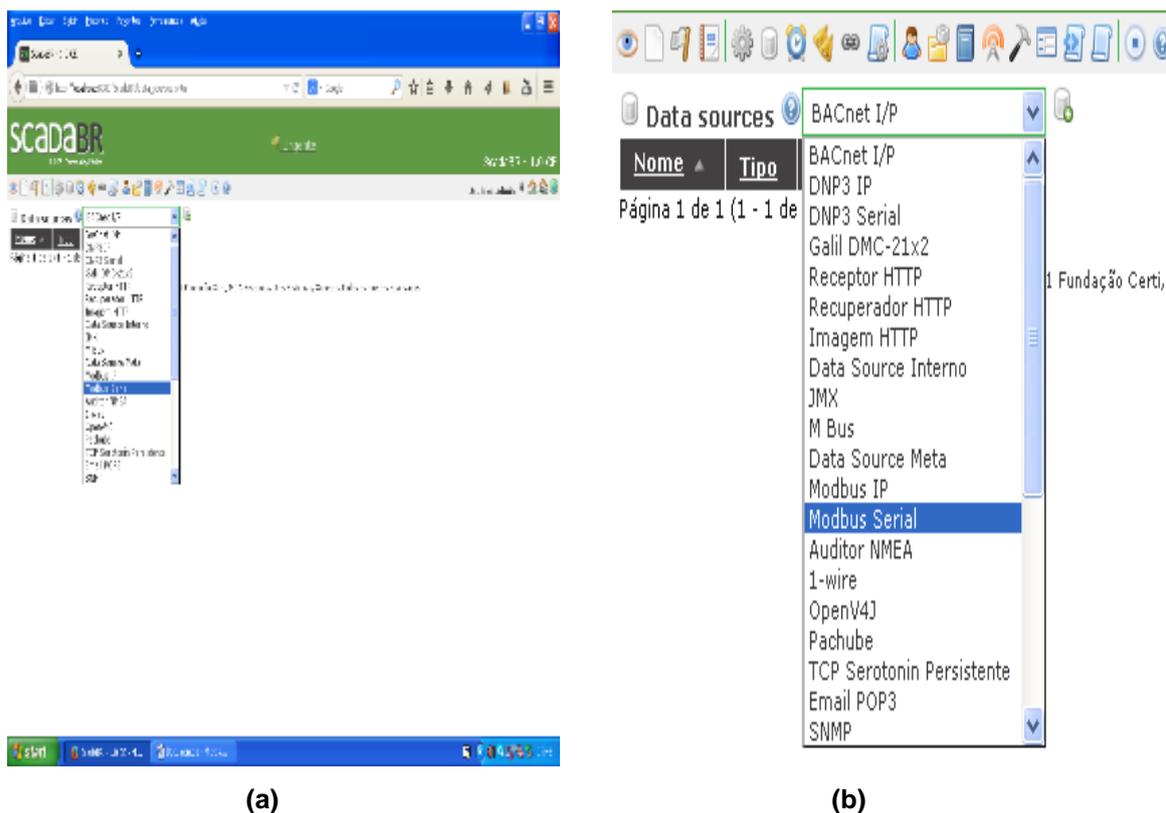


Figura 22 (a) e (b): Configurando Data Source
Fonte: Autoria própria (2016)

Dentro dos parâmetros de configuração foram configurados alguns itens necessários para conexão entre o *software* e a placa Arduino Uno, dentre os quais:

- Porta COM; esta foi configurada como 5 sem qualquer dificuldade pois o *software* lista todas as portas COM do *laptop* e a única utilizada pertencia a placa microcontrolada.
- Codificação; selecionado RTU.
- *Baud Rate*; selecionado 57600 como configurado na placa Arduino.
- *Data bits*, *Stop bit* e *Parity*: foi selecionado respectivamente 8, 1 e nenhuma.

Demais parâmetros foram utilizadas configurações *default* do *software*, essas configurações são apresentadas na Figura 23.

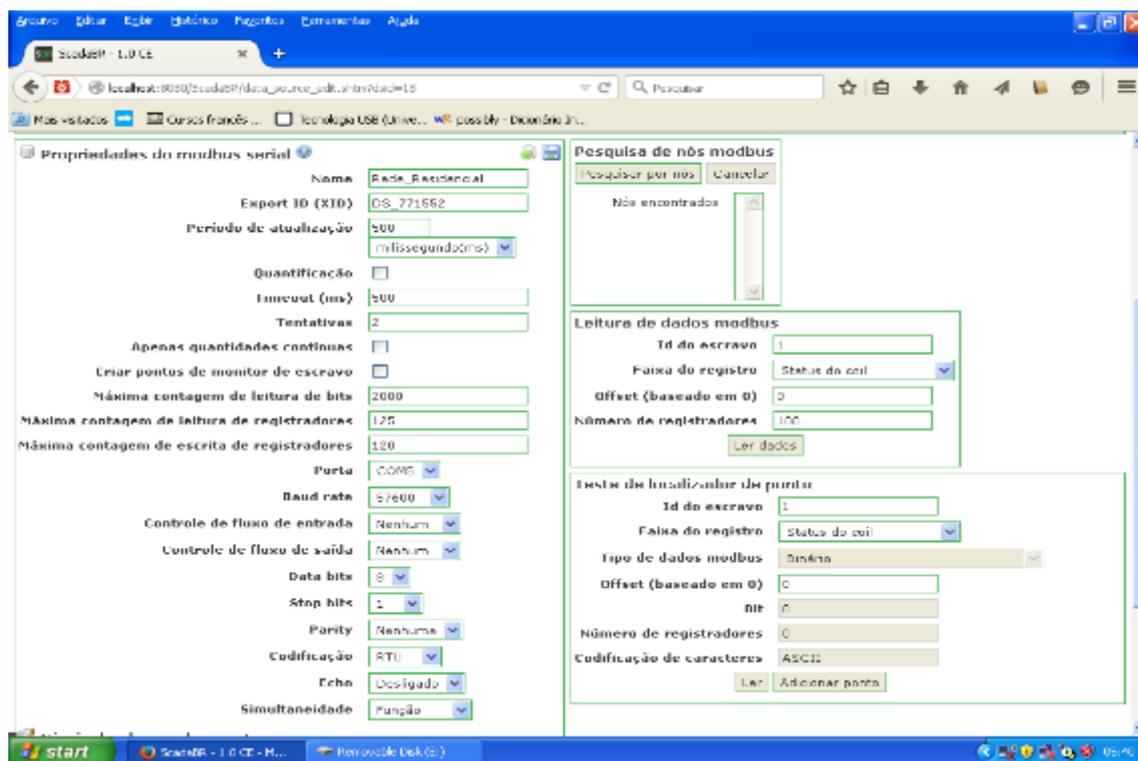


Figura 23: Configurando parâmetros de comunicação
Fonte: Aatoria Própria (2016)

O protocolo Modbus é protocolo mestre-escravo e, portanto, nesse caso o software ScadaBR se comporta como mestre e faz todas as solicitações de informações e comandos aos escravos. A placa Arduino Uno, nesse sistema, trata-se de um escravo e para a troca de dados com o mestre é necessário um endereço que o identifique. Este pode ser de 1 até 247 o número máximo de escravos conectados a esse mestre. Para identificar a placa Arduino foi feito uso da função pesquisa por nós, que identificou o *hardware* conectado com o endereço 1 que havia sido configurado previamente na placa microcontrolada. Com esse dado foi possível utilizando a função leitura de dados apresentar a variável transmitida pela placa Arduino Uno. A Figura 24 apresenta a função Pesquisa por nós Modbus, que ao ser selecionada mostra o endereço dos escravos disponíveis na rede. Com o endereço dos equipamentos disponíveis podemos pesquisar as variáveis desses dispositivos. Para isso temos a função Leitura de dados Modbus, basta selecionar o endereço, tipo da variável, *Offset* (variáveis com endereços menores não são pesquisadas) e número de registradores (são pesquisadas as variáveis até esse valor excetuando-se na faixa de *Offset*).

The image shows two windows from a software application. The top window, titled "Pesquisa de nós modbus", contains a search bar with "Pesquisar por nós" and "Cancelar" buttons. Below it, the text "Pesquisa completada" is displayed in red, followed by "Nós encontrados" and a spinner control showing the value "1". The bottom window, titled "Leitura de dados modbus", has several input fields: "Id do escravo" with the value "1", "Faixa do registro" with a dropdown menu showing "Registrador holding", "Offset (baseado em 0)" with the value "0", and "Número de registradores" with the value "1". A "Ler dados" button is located below these fields. At the bottom left of this window, the text "0 ==> 0000" is visible. A vertical scrollbar is on the right side of the window.

Figura 24: Pesquisando registradores
Fonte: Autoria própria (2016)

Finalmente bastou salvar a variável criada, essa foi nomeada Tensão_Rede. Na Figura 25 é apresentado o campo para salvar a variável. O nome é atribuído pelo usuário do *software*, já o *Export ID* é uma identificação gerada pelo sistema. Os campos multiplicador e aditivo servem para realizar essas operações matemáticas se houver necessidade. A caixa de seleção configurável deixa disponível a opção de comandar esse registrador.

Detalhes do data point	
Nome	Tensão_Redde
Export ID (XID)	DP_487202
Id do escravo	1
Faixa do registro	Registrador holding
Tipo de dados modbus	Inteiro de 2 bytes sem sinal
Offset (baseado em 0)	0
Bit	0
Número de registradores	0
Codificação de caracteres	ASCII
Configurável	<input type="checkbox"/>
Multiplicador	1
Aditivo	0

Figura 25: Variável criada
Fonte: Autoria própria (2016)

O ScadaBR possui o ícone *Watch list* que mostra as variáveis e é representado por um olho na barra de ferramentas do *software*. Acessando esse ícone foi possível observar a leitura da variável de tensão sendo atualizada. A Figura 26 mostra essa tela.

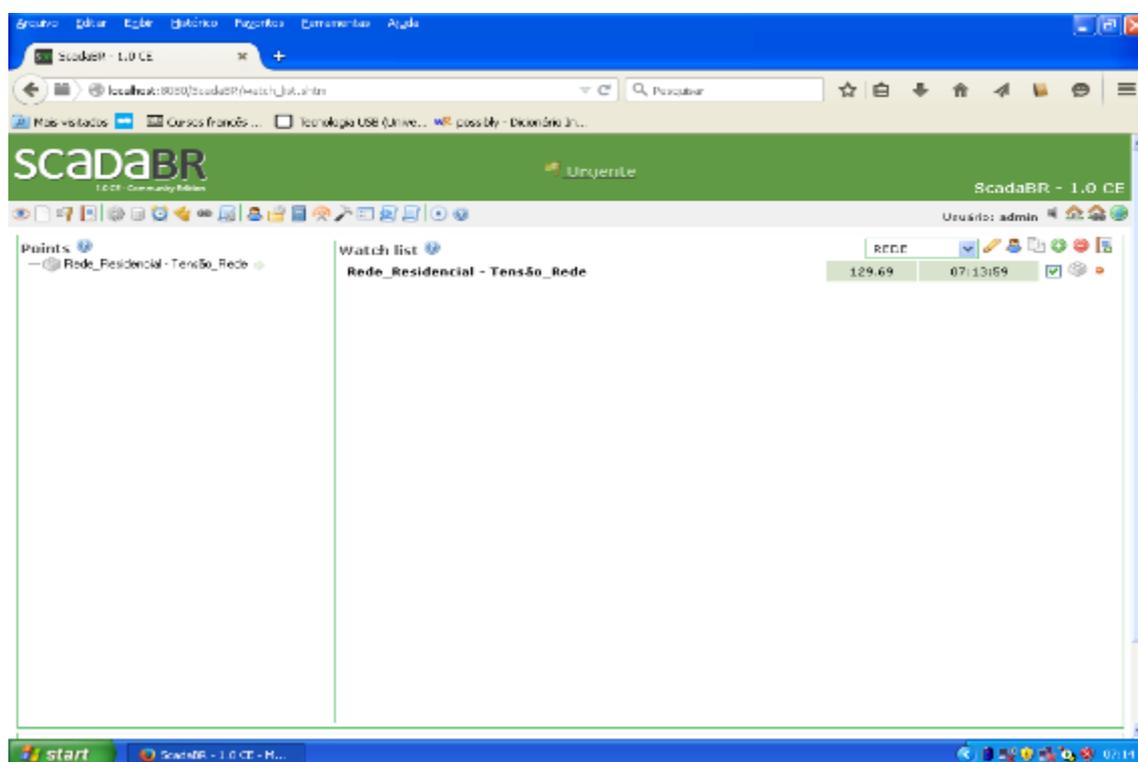


Figura 26: Watch list
Fonte: Autoria própria (2016)

Para tornar a visualização facilitada foi configurado alarme para mostrar o evento de desenergização da rede e retorno da alimentação. Para isso acessando o ícone detalhes do *data point* e editando-o, foi adicionado alarme com percepção de quando o valor de tensão da rede ficar abaixo de 80 Volts CA e seu retorno quando a tensão ultrapassasse 90 Volts. Na figura 27 é possível verificar a tela de detectores de eventos sendo adicionada. Foi adicionado dois detectores com sensibilização para a desenergização da rede assim como para o retorno do fornecimento, o nível do alarme foi estabelecido como urgente para a interrupção do fornecimento de energia por ser o objeto principal do monitoramento, o retorno apenas informação.

Ir para: REDE - Tensão_Redre ▾

Detectores de eventos ?

Tipo	Limite superior ▾ 📌
Tipo	Detector de limite inferior
Export ID (XID)	PED_578996
Alias	Interrupção do fornecimento de energia
Nível de alarme	Urgente ▾ 📌
Limite inferior	80
Duração	3 ▾ segundo(s) ▾
Tipo	Detector de limite superior
Export ID (XID)	PED_626206
Alias	Retorno do fornecimento
Nível de alarme	Informação ▾ 📌
Limite superior	90
Duração	1 ▾ segundo(s) ▾

Figura 27: Configuração detectores de eventos
Fonte: Autoria própria (2016)

Agora podemos visualizar as interrupções de fornecimento com seus horários para facilitar o entendimento das interrupções, na Figura 28 agora é apresentado a *watch list* com os alarmes.

Watch list 

(sem nome)     

REDE - Tensão_Rede 130,79Volts(CA) 07:00:43   

-  07:00:41 - Retorno do fornecimento  
-  07:00:36 - Interrupção do fornecimento de energia  
-  06:24:05 - Retorno do fornecimento  
-  06:22:19 - Interrupção do fornecimento de energia  
-  06:17:00 - Retorno do fornecimento  
-  06:13:46 - Interrupção do fornecimento de energia  
-  Jun 15 23:08 - Retorno do fornecimento  

Figura 28: Eventos na Watchlist
Fonte: Autoria própria (2016)

5 RESULTADOS

Após todas as etapas necessárias para o correto funcionamento do sistema, foi necessário realizar alguns testes para garantir que a finalidade de monitorar a rede elétrica residencial da unidade consumidora seria possível com o sistema. O primeiro teste consistiu em que o sistema ficasse ligado por aproximadamente 4 horas conectado à rede que estava energizada para observar se a leitura ocorreria de forma satisfatória. Ao final do período pode ser observado o gráfico com a tensão caindo próximo a zero pois foi desligado o sensor da rede de energia elétrica. A curva gerada pelo sistema supervisorio que mostra o seu funcionamento está representada na Figura 29.



Figura 29: Gráfico ScadaBR
Fonte: Autoria própria (2016)

O passo seguinte foi a simulação de eventos na rede elétrica. Foram simulados desligamentos retirando a fonte de alimentação da rede residencial do sensor. Essa simulação reproduz uma série de desligamentos na rede da unidade consumidora. Contudo tais desligamentos não seriam considerados no indicador DIC pois o maior desligamento ocorrido foi as 14:07 com retorno da tensão foi as 14:10, portanto segundo critério para contabilização para o indicador DIC os desligamentos

devem ser superiores a 3 minutos. A Tabela 4 apresenta os eventos simulados e os resultados obtidos.

Tabela 5: Testes de leitura

Evento simulado	Horário	Resultado
Rede energizada	13:57	OK
Rede desenergizada	13:59	OK
Rede energizada	14:01	OK
Rede desenergizada	14:03	OK
Rede energizada	14:05	OK
Rede desenergizada	14:07	OK
Rede energizada	14:10	OK
Rede desenergizada	14:12	OK

Fonte Autoria própria (2016)

A visualização gráfica é muito mais amigável ao usuário e facilmente gerada pelo sistema. Na Figura 30 é apresentado o gráfico gerado pelo *software* ScadaBR com os eventos simulados.

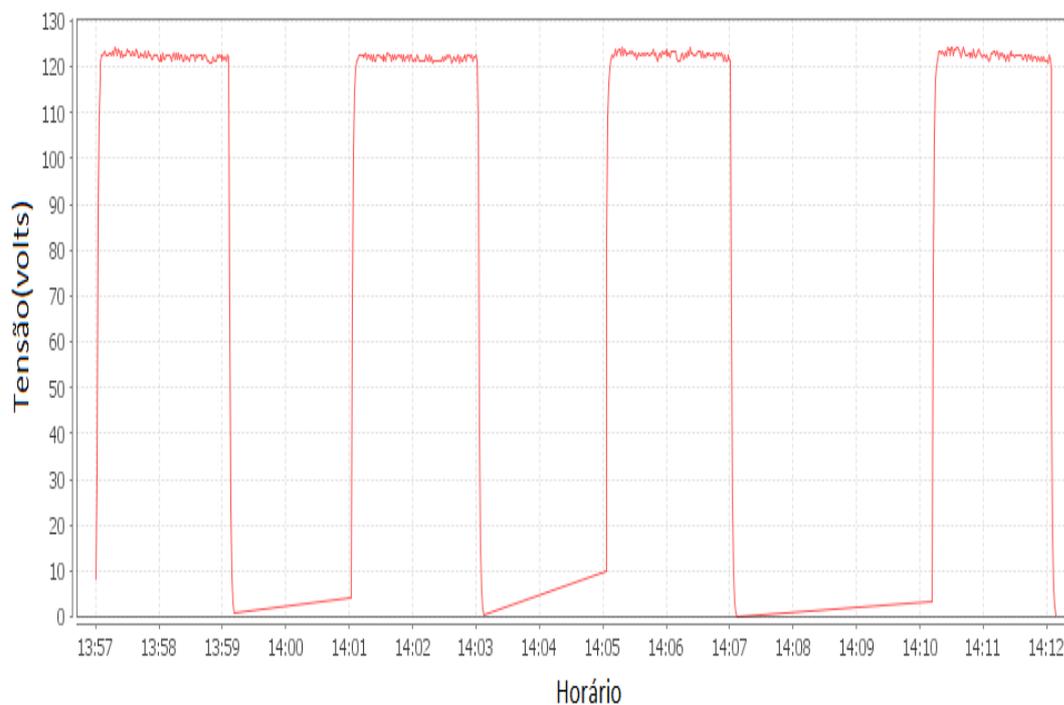


Figura 30: Gráfico dos desligamentos simulados
Fonte: Autoria própria (2016)

Neste caso foi simulado um terceiro desligamento de período maior que 3 minutos que seria computado no indicador. Na Figura 31 são apresentados os eventos gerados pelos alarmes com seus horários respectivos.



Figura 31: Eventos de interrupção
Fonte: Autorial própria (2016)

O desligamento aconteceu às 20:30:51 com retorno do fornecimento as 20:42:37, portanto 11 minutos e 46 segundos ou 0,196 hora, arredondando 0,20 que comporiam o indicador DIC do mês corrente. Devemos lembrar que esse indicador é expresso em horas com duas casas decimais depois da vírgula. Na Figura 32 está demonstrado o gráfico gerado pelo ScadaBR desse evento.

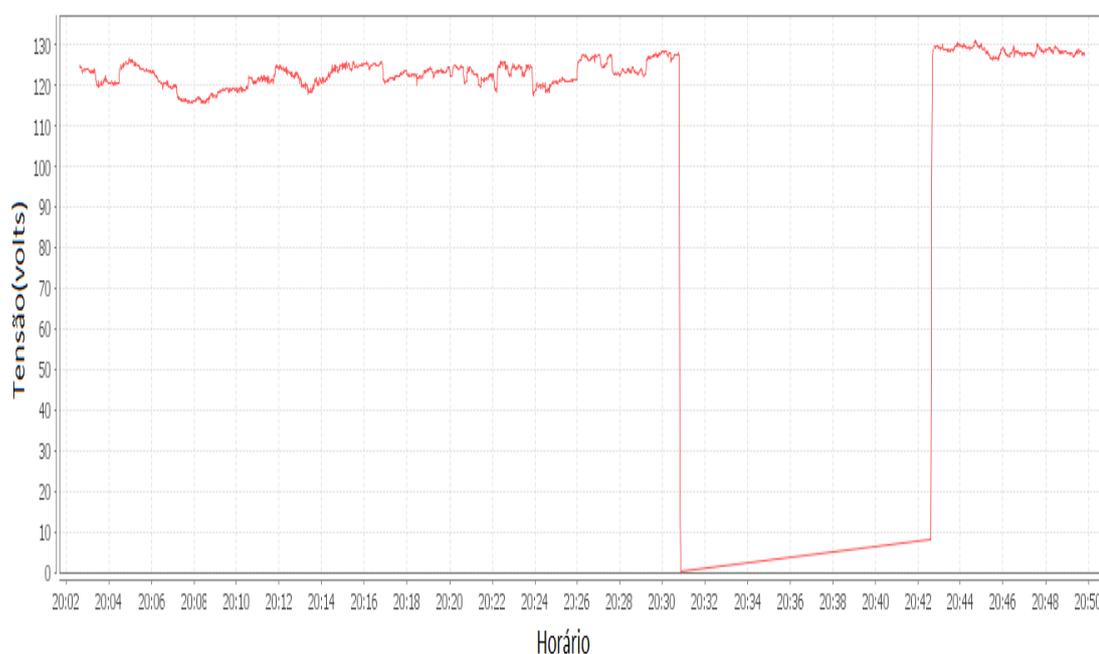


Figura 32: Simulação desligamento
Fonte: Autorial própria (2016)

Foram realizadas a utilização do *software* supervisor ScadaBr com sua configuração para leitura de dados encaminhados pela Placa Arduino Uno.

A instalação da placa Arduino foi bem sucedida, bem como o sensor para leitura da tensão da rede elétrica.

Os testes realizados para comprovar o funcionamento do sistema com interrupções simuladas foram satisfatórios.

O monitoramento da unidade consumidora residencial através de uma um microcontrolado ATmega328 na plataforma Arduino Uno integrado ao *software* supervisor ScadaBr foi executada com êxito, possibilitando a visualização de gráficos e relatórios das interrupções de fornecimento ocorridas na unidade consumidora.

Para demonstrar o cálculo do valor a ser ressarcido pela empresa fornecedora em caso de violação do indicador DIC vamos utilizar os dados da interrupção de 11 minutos e 47 segundos apresentado na figura 32, para apresentar como é aplicada a fórmula do cálculo do valor de ressarcimento ao cliente. Devemos então converter esse valor de tempo que habitualmente expressamos no formato hora:segundos para hora, o valor que auferimos no desligamento de 11 minutos e 47 segundos, ou seja 0,196 ou arredondando 0,20 hora. Usaremos os dados dos indicadores de qualidade de uma fatura real como exemplo apresentado na Figura 33.

Indicadores de Qualidade					
Conjunto: SABARA Mês Ref.: 02/2016					
	DIC	FIC	DMIC	DICRI	EUSD (R\$)
Realizado:	0,00	0,00	0,00	-	31,17
Limite Mensal:	4,83	3,17	2,69	12,22	
Limite Trimestral:	9,67	6,35	-	-	
Limite Anual:	19,34	12,70	-	-	
Tensão Contratada: 127 volts Limite Adequado de Tensão: 117 a 133					
O não cumprimento dos indicadores DIC, FIC, DMIC e DICRI definidos pela ANEEL resulta em compensação financeira ao consumidor pela concessionária no faturamento. É direito do consumidor solicitar a apuração destes indicadores a qualquer tempo.					

Figura 33: Indicadores para cálculo
Fonte: Autoria própria (2016)

Como podemos observar o valor do indicador apurado no exemplo não faria jus ao ressarcimento pois não viola o limite mensal de 4,83.

No entanto para conseguirmos demonstrar o cálculo do ressarcimento suponhamos que esse desligamento foi um dos ocorridos no mês apurado. Vamos atribuir aleatoriamente para exemplificar outra ocorrência de 2 minutos e mais uma terceira de 5 horas. Na tabela 5 apresentamos esses dados para melhor entendimento do cálculo do DIC.

Tabela 6: Apuração valor DIC mensal

Sequência de Interrupção	Interrupções (hora)	Observação
1	0,20	
2		Interrupção < que 3 minutos (2 minutos).
3	5,00	
TOTAL	5,20	

Fonte: autoria própria (2016)

Com o valor apurado de 5,20 para o DIC mensal extrapolou-se o limite de 4,83, portanto gerando direito de ressarcimento ao cliente. Agora vamos aplicar a fórmula, onde ainda nos referenciamos aos valores da Figura 33 temos:

$$Valor = \left(\frac{DICv}{DICp} - 1 \right) DICp \times \frac{EUSDm\u00e9dio}{730} \times kei$$

- DICv = 5,20;
- DICp = 4,83;
- EUSD médio = 31,17 (observemos que se trata da média apurada no ano corrente, o expresso na fatura trata-se do EUSD do mês, para simplificar adotaremos esse valor como EUSDm\u00e9dio).

- Kei = 15 por se tratar de consumidor de baixa tens\u00e3o.

$$Valor = \left(\frac{5,20}{4,83} - 1 \right) 4,83 \times \frac{31,17}{730} \times 15$$

$$Valor = 0,24$$

Portanto o ressarcimento seria de R\$ 0,24 em forma de desconto na fatura de energia do cliente e poderia ser creditado at\u00e9 dois meses ap\u00f3s a apura\u00e7\u00e3o.

6 CONCLUSÃO

Com esse trabalho pretendeu-se disponibilizar informações obtidas em uma rede de energia elétrica residencial monofásica através de um banco de dados. Esses dados armazenados em um computador portátil. Gerar relatórios das interrupções ocorridas com seus respectivos períodos. Também demonstrar como utilizar esses dados para apurar o indicador de qualidade individual DIC, que pode gerar multas a fornecedora de energia em favor do cliente na forma de descontos na fatura. Aplicar a forma de cálculo do ressarcimento com os dados obtidos.

Para isso foram utilizados recursos tecnológicos com baixo custo total. O *software* supervisor *Open Source ScadaBR* não gerou custo ao projeto, enquanto que o a plataforma Arduino Uno, sensor de tensão e demais recursos totalizaram R\$ 74,30.

Com sucesso foi realizado o monitoramento e armazenamento de dados e geração de relatórios. Para isso algumas etapas foram realizadas como instalação e configuração do ScadaBr, conexão do sensor ao microcontrolador, instalação da placa Arduino Uno e ambiente de desenvolvimento (IDE). Conseguiu-se a integração dos recursos fazendo com que as informações obtidas fossem encaminhadas no formato do protocolo Modbus, pela porta USB, da placa Arduino ao computador com o *software* supervisor instalado.

Com o monitoramento das interrupções realizado com sucesso na rede elétrica residencial, a apuração das interrupções através de dados armazenados no banco de dados do *software* ScadaBR. Foi possível demonstrar e aplicar os critérios da composição do indicador de qualidade individual DIC. Mostrou-se aplicável o uso dos dados coletados para confronto com valores apresentados na fatura de energia haja vista que o programa possibilita a retirada de informações selecionando o período desejado gerando-se gráficos com os dados desejados.

Como possíveis trabalhos futuros pode-se implementar monitoramento de outros indicadores de qualidade individuais como o indicador Frequência de Interrupção por unidade Consumidora (FIC) que demonstra a quantidade de interrupções ocorridas, como também o indicador Duração Máxima de Interrupção contínua por unidade Consumidora (DMIC) que demonstra a duração da interrupção mais longa ocorrida na unidade. Existe também a possibilidade de gerência do consumo da energia elétrica da unidade consumidora.

REFERÊNCIAS

AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. **Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional – PRODIST Módulo 8 – Qualidade da Energia Elétrica**. Disponível em: <http://www.aneel.gov.br/modulo-8>. Acesso em 28 maio 2016.

ALBUQUERQUE, Pedro Urbano Braga de; ALEXANDRIA, Auzuir Ripardo de. **Redes industriais**. São Paulo: Ensino Profissional, 2009.

CASILLO, Danielle. **Controle e programação na automação**. Disponível em: < www2.ufersa.edu.br>. Acesso em: 20 maio 2015.

COELHO, Marcelo Saraiva. **Padrão modbus**. Disponível em: <<http://www.ebah.com.br/content/ABAAAgKooAJ/rede-modbus>>. Acesso em: 19 mar.2015

COELHO, Marcelo Saraiva. **Apostila de sistemas Supervisórios**. Disponível em: <<http://www.ebah.com.br/content/ABAAAA-eoAK/apostila-sistemas-supervisorios-revisao-3-2009-pdf-ifsp-automacao>>. Acesso em: 21 mar.2015

EVANS, Martin; NOBLE, Joshua; HOCHENBAUM, Jordan. **Arduino em ação**. São Paulo: Novatec Editora, 2013.

FUNDAÇÃO CERTI. **Manual ScadaBR**. Disponível em: <<http://www.scadabr.com.br/?q=downloads>>. Acesso em 03 mar.2015

LABORATÓRIO DE GARAGEM. **Padrão Modbus**. Disponível em: <labdegaragem.com/forum/attachment/download>. Acesso em: 15 maio 2015.

LANGBRIDGE, James A. **Arduino Sketches**. Indianapolis: John Wiley & Sons INC, 2015.

MORAES, Cícero Couto de; CASTRUCI, Plínio de Lauro. **Engenharia de Automação Industrial**. Rio de Janeiro: LTC, 2013.

MCRBERTS, Michael. **Arduino Básico**. São Paulo: Novatec, 2011.

MCRBERTS, Michael. **Arduino Básico**. 2 ed. São Paulo: Novatec, 2015.

NOVAELETRONICA. **Pinagem de Conectores USB**. Disponível em <<http://blog.novaeletronica.com.br/pinagem-de-conectores-usb/>>. Acesso em: 10 maio 2016.

REAL TIME AUTOMATION. Disponível em: <<http://www.rtaautomation.com/technologies/modbus-rtu/>>. Acesso em: 16 jun 2016>.

SILVA, Ana P.G. da.; SALVADOR, Marcelo. **O que são sistemas supervisórios?** Disponível em: <http://www.wectrus.com.br/artigos/sist_superv.pdf>. Acesso em: 21fev.2015.

SEIXAS FILHO, Constantino. **Deteção de erros**. Disponível em: <www.ufmg.edu.br>. Acesso em: 19 maio 2015.

TANEMBAUM, Andrew S. **Redes de computadores**. São Paulo. Editora Campus, 2009.

TORRES, Gabriel . **Hardware Curso Completo**. 4. ed. Rio de Janeiro: Axcel Books do Brasil Editora, 2001.

ZELENOVSKI, Ricardo; MENDONÇA, Alexandre. **PC: um Guia prático de Hardware e Interfaceamento**. 3.ed. Rio de Janeiro . MZ editora LTDA, 2002.