

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO SUPERIOR DE TECNOLOGIA EM ELETRÔNICA

JOSÉ PANTOJA NETO

AQUISIÇÃO DE DADOS UTILIZANDO A TECNOLOGIA GPRS

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2013

JOSÉ PANTOJA NETO

AQUISIÇÃO DE DADOS UTILIZANDO A TECNOLOGIA GPRS

Trabalho de Conclusão de Curso de Graduação, apresentado à disciplina de Trabalho de Diplomação do Curso Superior de Tecnologia em Eletrônica do Departamento de Eletrônica – DAELN - da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Luiz Fernando Copetti

CURITIBA
2013

TERMO DE APROVAÇÃO

JOSÉ PANTOJA NETO

AQUISIÇÃO DE DADOS UTILIZANDO A TECNOLOGIA GPRS

Este trabalho de conclusão de curso foi apresentado no dia 14 de dezembro de 2011, como requisito parcial para obtenção do título de Tecnólogo em Eletrônica: Modalidade Automação Industrial, outorgado pela Universidade Tecnológica Federal do Paraná. O aluno foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Luis Carlos Vieira
Coordenador de Curso
Departamento Acadêmico de Eletrônica

Prof. Dr. Sérgio Moribe
Professor responsável pela Atividade de Trabalho de Conclusão de Curso
Departamento Acadêmico de Eletrônica

BANCA EXAMINADORA

Prof.: Gabriel Kovalhuk
UTFPR

Prof.: Wagner Leitão
UTFPR

Prof. Luiz Fernando Copetti
Orientador - UTFPR

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

RESUMO

PANTOJA NETO, José. **Aquisição de dados utilizando tecnologia GPRS**. 2011. 40 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Eletrônica), Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2013.

Este trabalho tem como objetivo utilizar a tecnologia GPRS com a finalidade de atualizar dados enviados por um equipamento de aquisição remoto. Em virtude da necessidade de comunicação entre pontos remotos de sistemas automatizados, onde não é possível a transmissão de dados via cabos, a tecnologia sem fio GPRS mostra-se como solução ideal para este tipo de problema. Alguns protocolos serão utilizados para a transferência de dados, enfatizando o Modbus, por possuir uma estrutura de mensagem aberta. Neste estudo, serão levantadas características da tecnologia GPRS, bem como dos demais protocolos utilizados.

Palavras-Chave: Protocolo Modbus. Sistemas Distribuídos. GPRS.

ABSTRACT

PANTOJA NETO, José. **Data acquisition using GPRS technology**. 2011. 40 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Eletrônica), Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2013.

This study aims to use GPRS in order to update data sent from a remote acquisition equipment. Because of the need for communication between remote locations of automated systems, where it is not possible to transmit data via cable, wireless technology GPRS shows up as an ideal solution for this problem. Some protocols will be used for data transfer, emphasizing the Modbus, by having a structure of open message. In this study, will be raised features GPRS technology, as well as other protocols used.

Keywords: Modbus protocol. Distributed Systems. GPRS.

LISTA DE ILUSTRAÇÕES

Figura 1 - Processo de requisição e resposta entre cliente e servidor	14
Figura 2 - Representação de servidores com papel de clientes	14
Figura 3 - Representação das sete camadas e seus relacionamentos	17
Figura 4 - Camadas adicionando ao pacote informações de controle	17
Figura 5 - Interfaces entre cliente e servidor segundo o padrão OPC	23
Figura 6 - Bancada de teste com o CLP conectado ao modem GPRS	28
Figura 7 - Topologia do projeto.....	28
Figura 8 - Relação Cliente-Servidor e Mestre-Escravo	29
Figura 9 - Comunicação entre os nós da rede.....	31
Figura 10 - Configurando portas TCP.....	31
Figura 11 - Configurando portas Modbus TCP.....	32
Figura 12 - Tela do aplicativo que controla o protocolo Modbus (Servidor).....	33
Figura 13 - Equipamentos de campo.....	34
Figura 14 - Comunicação entre o servidor Modbus e o supervisório	35
Figura 15 - Servidor Modbus e remota conectados	36
Figura 16 - Supervisório comunicando com servidor Modbus	37

LISTA DE QUADROS

Quadro 1 – Comandos Modbus	20
Quadro 2 - Pergunta comunicação Modbus	21
Quadro 3 - Resposta comunicação Modbus.....	21

LISTA DE TABELAS

Tabela 1 - Comparação de custo TCP versus UDP	30
--	-----------

LISTA DE ABREVIATURAS, SIGLAS E ACRÔNIMOS

ACK – Acknowledged
ADU – Application Data Unit
ASCII – American Standard Code for Information Interchange
CLP – Controlador Lógico Programável
CRC – Cyclic Redundancy Check
DNS – Domain Name System
DDNS – Dynamic DNS
ERP – Enterprise Resource Planning
FTP – File Transfer Protocol
GPRS – General Packet Radio Service
GSM – Global System Mobile
HTTP – Hypertext Transfer Protocol
ISO – International Organization for Standardization
IP – Internet Protocol
MES – Manufacturing Execution Systems
NACK – Negative Acknowledged
OEE – Overall Equipment Effectiveness
OLE – Object Linking and Embedding
OPC – Object Linking and Embedding for Process Control
OSI – Open Systems Interconnection
PDU – Protocol Data Unit
RTU – Remote Terminal Units
SCADA – Supervisory Control and Data Acquisition
SIM – Subscriber Identity Module
SFC – Shop Floor Control
SMTP – Simple Mail Transfer Protocol
TCP – Transmission Control Protocol
UDP – User Datagram Protocol

SUMÁRIO

1 INTRODUÇÃO.....	10
1.1 PROBLEMA.....	11
1.2 OBJETIVOS.....	11
1.2.1 Objetivo geral.....	11
1.2.1 Objetivos específicos.....	11
1.3 JUSTIFICATIVA.....	12
1.4 MÉTODO DE PESQUISA.....	12
1.5 FUNDAMENTAÇÃO TEÓRICA.....	13
1.5.1 Sistemas distribuídos.....	13
1.5.1.1 Modelo cliente-servidor.....	13
1.5.2 Características de um sistema distribuído.....	15
1.5.3 Protocolos de comunicação.....	15
1.5.4 Modelo OSI.....	16
1.5.5 Protocolo TCP/IP.....	18
1.5.6 DNS e DDNS.....	18
1.5.7 Protocolo Modbus.....	19
1.5.8 Tecnologias de comunicação sem fio.....	21
1.5.9 GPRS.....	22
1.5.10 Padrão de comunicação OPC.....	23
1.5.11 Sistema de supervisão (SCADA).....	24
1.5.12 CLP.....	25
2. DESENVOLVIMENTO.....	27
2.1 ARQUITETURA DO SISTEMA.....	27
2.2 COMUNICAÇÃO ENTRE OS DISPOSITIVOS.....	29
2.3 COMUNICAÇÃO GPRS.....	31
2.4 SERVIDOR MODBUS TCP.....	31
2.5 TELA DO SISTEMA SUPERVISÓRIO.....	33
2.6 TESTES.....	35
3 CONCLUSÃO.....	38
4 ANEXOS.....	40

1. INTRODUÇÃO

A necessidade de gerar resultados satisfatórios em intervalos de tempo mais curtos, aumento da velocidade de produção, diminuição de custos tem levado as indústrias à busca de novos meios de suporte à produção. A automação torna-se a melhor alternativa para resolução destes problemas porque ela pode gerenciar todo o processo industrial eliminando tempos ociosos do processo, diminuindo consumo de energia e matéria-prima. A busca por soluções, que melhorem o processo de produção e diminuam os custos, fazem com que novas tecnologias sejam criadas e novas soluções com as tecnologias já existentes sejam adaptadas para aplicações específicas a todo o momento.

A comunicação por meio de cabos para interligação dos equipamentos é utilizada tanto nos processos de produção de bens quanto nos processos de prestação de serviços. Há, porém, situações em que isto não é possível devido à distância entre os pontos ou pelo difícil acesso aos locais de aquisição de dados. Nestes casos, a comunicação sem-fio, especificamente a tecnologia GPRS, é a melhor alternativa para resolução deste problema. Esse tipo de comunicação tem como característica principal a comutação de pacotes de dados. Neste sistema, o canal só é utilizado quando se está enviando/recebendo dados e assim deixando o canal livre para outros usuários do serviço. Os *slots* são alocados conforme a demanda por pacotes a enviar e/ou receber. Além disso, possui a característica de cobrança por volume de dados transmitidos, ocasionando uma redução do custo mensal da comunicação, e elimina a utilização de cabos, reduzindo o preço das instalações.

Este projeto tem como finalidade demonstrar uma aplicação baseada na tecnologia GPRS, integrando o sistema através de um Controlador Lógico Programável (CLP), um modem GPRS e uma estação servidora de dados. Além disto, será implementado um *driver* Modbus para a comunicação entre a remota (CLP) e o aplicativo supervisório.

1.1 PROBLEMA

A necessidade de integrar quaisquer sistemas remotos (CLP's) distantes geograficamente de um determinado centro de controle (Servidor) com baixo custo deve-se à hostilidade do meio físico, caracterizada por diversos fatores como, elevadas temperaturas, excesso de água e interferências. Em alguns casos estes fatores tornam inviável a comunicação utilizando cabos tradicionais.

1.2 OBJETIVOS

1.2.1 Objetivo geral

Realizar a troca de dados entre um CLP, o qual controla os dispositivos de campo, e uma estação SCADA, utilizando a tecnologia sem fio GPRS, avaliando o desempenho e funcionamento desse tipo de comunicação.

1.2.1 Objetivos específicos

- Realizar um estudo sobre sistemas distribuídos e os protocolos de comunicação utilizados neste projeto;
- Definir uma arquitetura para o sistema;
- Selecionar os dispositivos adequados para a comunicação;
- Implementar um protocolo de comunicação de dados adequado, responsável pela transmissão de dados entre a remota (CLP) e a estação servidora de dados.
- Realizar testes de comunicação para verificar se os objetivos do sistema foram atingidos.

1.3 JUSTIFICATIVA

Dados os problemas existentes para realizar a troca de informações entre equipamentos remotos de uma rede e um centro de controle, a comunicação sem-fio é uma solução muito eficaz. A escolha da tecnologia sem-fio GPRS em detrimento de outras tecnologias sem-fio se deve as características do problema levantado, como grandes distâncias (o que impossibilita o uso de tecnologia sem-fio *bluetooth*) e alto custo (o que impossibilita o uso de tecnologia sem-fio wi-fi por rádio-modem pelo custo de instalação de torres, antenas e repetidores).

A tecnologia GPRS permite aos seus usuários estarem sempre conectados à rede, prontos a enviar e receber informações via *internet* de forma on-line e imediata. O acesso torna-se, assim, mais rápido, a transmissão de dados é processada com uma velocidade superior em relação à tecnologia GSM e os custos são reduzidos, pois a forma de valorização é baseada no volume de informações trafegadas e não pelo tempo real da transação efetuada. Enquanto o custo de 1 *kylobyte* na rede GPRS é de aproximadamente R\$0,004, na rede GSM o *kilobyte* custa aproximadamente R\$0,10. Assim, a tecnologia de telefonia móvel GPRS apresenta-se como uma boa solução para comunicar pontos remotos.

Verificando isso, o desenvolvimento de um *driver*, com especificação aberta, aparece como uma solução vantajosa e eficiente para a troca de informações utilizando esta tecnologia. A diferença de usar um *driver* com especificação aberta e criar um *driver* de comunicação totalmente novo é que é possível encontrar no mercado diversos equipamentos utilizando um *driver* de especificação aberta, ocasionando redução de custos, sendo também muito mais fácil sua configuração.

Com esse sistema, pretende-se provar a eficiência da tecnologia GPRS, levando em consideração seu custo da comunicação, confiabilidade e desempenho.

1.4 MÉTODO DE PESQUISA

Para a realização desta monografia, será adotada uma pesquisa exploratória, envolvendo estudos de campo e técnicas experimentais.

A pesquisa bibliográfica, direcionada em livros, artigos de revistas, *sites* especializados no assunto, será adotada como instrumento base para o estudo do projeto.

1.5 FUNDAMENTAÇÃO TEÓRICA

1.5.1 Sistemas distribuídos

Os sistemas distribuídos, segundo Coulouris (2006) são sistemas que consistem em várias unidades de processamento interligadas e com responsabilidades definidas seguindo padrões para distribuição de dados e da carga de trabalho, buscando, com isso, a otimização da realização de um processo comum.

O principal objetivo de um sistema distribuído consiste em conectar os usuários e os recursos de forma transparente, aberta e escalável (estar preparado para crescer), compartilhando os recursos. Geralmente estas combinações se tornam mais tolerantes a falhas e mais poderosas do que um computador com processamento único. Segundo Simomura (2009), podemos afirmar que a *internet* é um sistema distribuído porque temos um conjunto de microcomputadores independentes que interligam-se mutuamente com o objetivo de compartilhar recursos.

1.5.1.1 Modelo cliente-servidor

Ainda segundo Coulouris (2006), “é a arquitetura mais citada quando os sistemas distribuídos são discutidos”. Este modelo diferencia dois tipos de processos em uma rede, um cliente e um servidor. De modo geral, um servidor opera de modo passivo, processando as requisições e oferecendo os resultados ao cliente que por sua vez tem o papel ativo, pois é ele quem faz a solicitação de serviço ao servidor. A Figura 1 mostra um diagrama que descreve o processo de solicitação e resposta, em que o servidor aguarda a iniciação do cliente para iniciar o processamento.

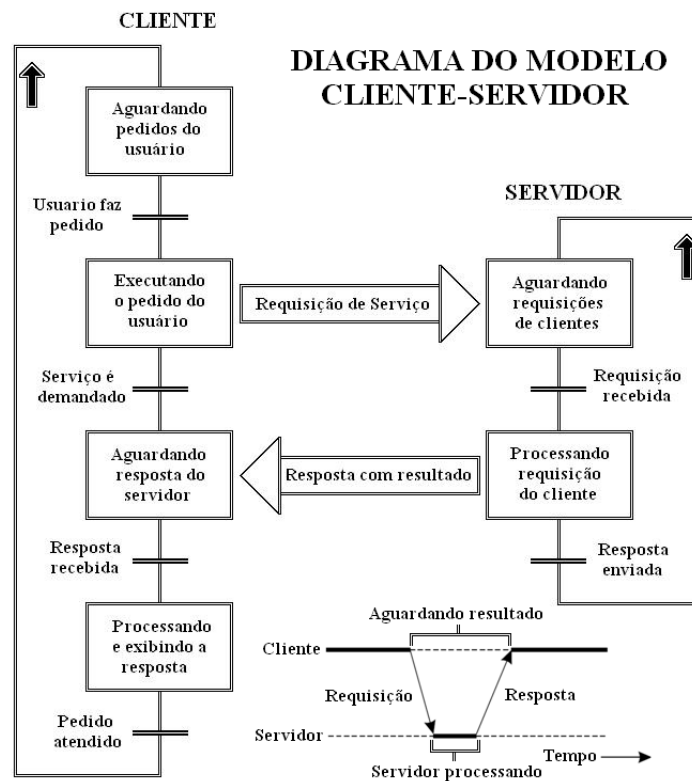


Figura 1 - Processo de requisição e resposta entre cliente e servidor
Fonte: Tanenbaum (2007)

Assim como mostra a Figura 2, um servidor pode fazer o papel de cliente perante outros servidores. Isso faz com que várias capacidades de uma rede possam ser mais bem distribuídas entre todos os clientes que são, geralmente, os usuários finais.

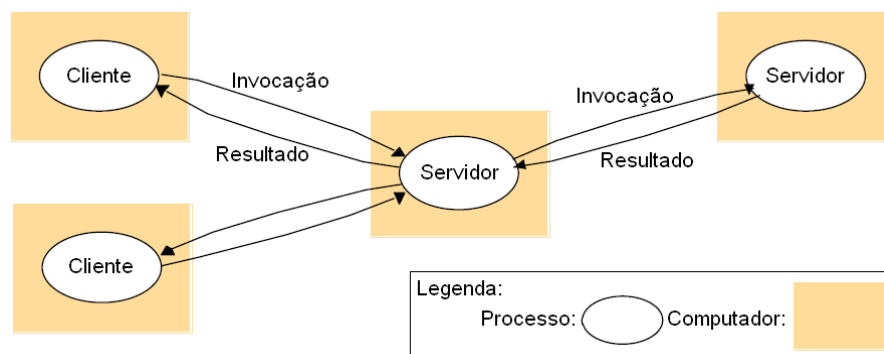


Figura 2 - Representação de servidores com papel de clientes
Fonte: Couloris (2006)

1.5.2 Características de um sistema distribuído

Coulouris (2006) apresenta as propriedades fundamentais que caracterizam um sistema distribuído, essas são:

- **Compartilhamento de Recursos:** é uma característica muito importante para os sistemas distribuídos. O alto nível de tecnologia de compartilhamento elimina a necessidade de um processador único e cria possibilidade da divisão de tarefa entre as várias partes do sistema assim como o fácil acesso a multiusuários e recursos remotos, tais como impressoras, páginas da *internet*, dados contidos em um servidor.
- **Escalabilidade:** um sistema distribuído é dito escalável quando possui potencial para ser expandido, ou seja, o número de nós pode ser aumentado sem que o seu desempenho piore.
- **Concorrência:** habilidade do sistema de compartilhar recursos entre diversos processos.
- **Tolerância a falhas:** através da redundância, os sistemas distribuídos se tornam mais tolerante a falhas e com isso aumentam a confiabilidade. Isto equivale à diminuição da probabilidade de que uma falha interrompa um processo do sistema.

1.5.3 Protocolos de comunicação

Os protocolos são regras de comunicação que encapsulam uma linguagem de modo que possibilite uma conexão, comunicação ou transferência de dados entre dois sistemas. Assim, a comunicação entre dois pontos tem que ser feita de forma padronizada e funcional. Além de garantir a integridade da mensagem o protocolo também cuida do endereçamento, pois, em muitos casos, existem mais de dois nós na rede. Neste caso é necessário que, na mensagem enviada pelo nó 1, exista o endereço do destinatário do nó 2 para que o nó 3 não perca tempo de processamento recebendo uma mensagem que não lhe diz respeito.

Para detecção de erros na transmissão, são utilizados em alguns protocolos, um ou mais *bytes* gerados a partir de algoritmos que envolvem os *bytes* da própria mensagem protocolada. Ao receber a mensagem, esses *bytes* são recalculados

utilizando o mesmo algoritmo e comparados com aqueles que vieram na mensagem. Caso não coincidam, é solicitada ao remetente a retransmissão dos dados.

Ao receber um pacote e verificar que sua integridade não foi afetada durante a transmissão, o receptor envia uma mensagem ao remetente dizendo que a mensagem foi entregue com sucesso (ACK). Caso haja um erro, envia uma mensagem negando o entendimento do pacote (NAK). A cada pacote enviado, o emissor numera esse pacote e dispara um temporizador, se a confirmação de recebimento da mensagem (ACK) não chegar e o tempo se esgotar, a mensagem é retransmitida. Quando o receptor recebe a mensagem e verifica que esta duplicada, ele descarta a mensagem e reenvia à confirmação de recebimento.

Este processo é chamado de cálculo de CRC (*Cyclic redundancy check*). Serão utilizados cálculos padrões de CRC para este trabalho.

Em muitos casos, para que tipos diferentes de sistemas se comuniquem, é utilizado um protocolo encapsulado dentro do outro. Temos como exemplo o protocolo Modbus sobre TCP/IP, que é transmitido através de redes padrão *ethernet* pelo protocolo TCP/IP e tem encapsulado dentro de si o protocolo Modbus.

1.5.4 Modelo OSI

Devido à necessidade de interconexão de equipamentos e processos de uma grande variedade de projetos de diferentes fabricantes, a ISO (*International Organization for Standardization*) decidiu criar um padrão único que servisse como base para os fabricantes na hora de criar seus protocolos, o modelo OSI (*Open Systems Interconnection*).

O modelo OSI divide-se em uma estrutura de sete camadas independentes, como mostra a Figura 3, onde cada uma delas é definida em função dos serviços que provem para a camada de nível imediatamente superior. Cada camada comunica-se apenas com as camadas imediatamente acima e abaixo dela.

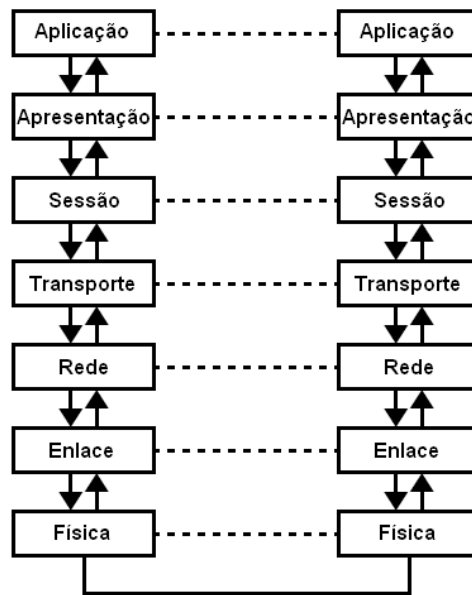


Figura 3 - Representação das sete camadas e seus relacionamentos
Fonte: Farias (2006)

A Figura 4 mostra que em uma transmissão de dados, a camada aplicação inicia o processo transferindo os dados para uma camada imediatamente abaixo, ao receber esses dados, a camada agrega a esses dados uma informação de controle que será interpretada pela mesma camada do lado do receptor e envia esta camada para um nível abaixo.

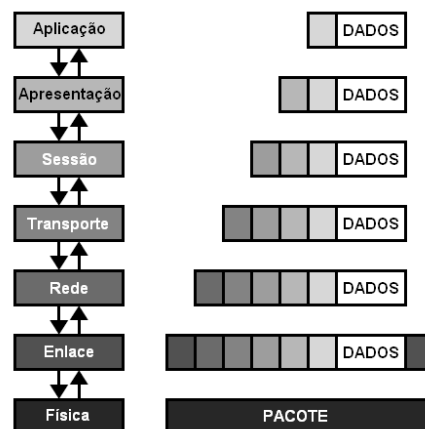


Figura 4 - Camadas adicionando ao pacote informações de controle
Fonte: Adaptado de Sverzut (2005)

Nota-se que na camada de enlace é adicionado um conjunto de controle a última posição do pacote, esse conjunto de bytes, também conhecido como CRC

(*Cyclic Redundancy Check*) é responsável pela verificação da integridade da mensagem.

1.5.5 Protocolo TCP/IP

De acordo com Sverzut (2005) o TCP/IP é na verdade, um conjunto de protocolos, que está intimamente associado à *Internet*. Seus dois principais protocolos são o TCP (*Transmission Control Protocol*) e o IP (*Internet Protocol*).

O protocolo TCP mostra-se confiável, pois permite a recuperação de pacotes perdidos, a eliminação de pacotes duplicados, a recuperação de dados corrompidos, e pode recuperar a ligação em caso de problemas no sistema e na rede. A aplicação, por ser orientada à conexão, envia um pedido de conexão para o destino e usa a conexão para transferir os dados.

O IP é um protocolo universal que proporciona a comunicação entre *hosts* ligados em rede. Ele oferece um serviço de datagramas não confiável; ou seja, os pacotes vêm sem garantias. Podem chegar desordenados (comparados com outros pacotes enviados entre os mesmos nós), podem também chegar duplicados, ou podem ser perdidos por inteiro. Se a aplicação requer maior confiabilidade, esta é adicionada na camada de transporte.

Seu método de conferência de integridade é feito através de uma soma de *bits* de seu cabeçalho que levam os endereços relevantes para a mensagem além de alguns outros dados de controle. Com isto, qualquer mudança nos endereços é facilmente detectada e por não envolver todos os *bytes* do pacote é rapidamente calculado.

1.5.6 DNS e DDNS

O DNS (*Domain Name System* – Sistema de Nomes de Domínios) é um sistema que “resolve” (transforma) nomes de domínios em endereços de rede (IP’s). Esse serviço se localiza na camada de aplicação do protocolo TCP/IP

Um *site* da *Web* pode ser identificado de duas maneiras: pelo seu domínio ou pelo endereço IP. Porém, nomes de domínio são mais fáceis de ser lembrados

pelos usuários particulares e das empresas do que endereços IP. Esta é a principal função do DNS, traduzir o número de endereço IP para o nome do domínio.

Tomando como exemplo o site da UTFPR, cujo domínio é “utfpr.edu.br” e o endereço IP é 200.019.073.182. É muito mais fácil memorizar o nome do que seu endereço IP.

O DDNS (*Dynamic DNS* – DNS Dinâmico) é um sistema que permite a atualização em tempo real de informações sobre nomes de domínios situados em um servidor de nomes. O uso mais comum que se dá é permitir a atribuição de um nome de domínio de *internet* a um computador com IP dinâmico. Assim é possível conectar à máquina em questão sem a necessidade de estar rastreando o endereço IP a cada vez que a conexão à *internet* é refeita.

Para que isso seja possível, é necessária a utilização de um *software* servidor no computador com IP dinâmico. Alguns desses programas são o No-IP e DynDNS.

1.5.7 Protocolo Modbus

Segundo a Organização Modbus (2005), o protocolo Modbus foi criado em 1979 pela empresa Modicon com a finalidade de comunicar dados entre seus controladores. Desde o início esse protocolo tem a especificação aberta e pode ser utilizado em vários níveis de aplicação, desde o sensor até o supervisor. Por esses motivos é largamente utilizado atualmente, além de ser uma das soluções de rede mais baratas para utilização em Automação Industrial.

O Modbus é um protocolo Mestre/Escravo ou Cliente/Servidor, ou seja, a estação Mestre/Cliente envia mensagens aos Escravos/Servidores solicitando que enviem os dados ou envia sinais a serem escritos nos Escravos/Servidores. Caso o escravo tenha informações que devam ser enviadas a outros escravos, este espera que o mestre lhe faça um requerimento para que então o mestre repasse os dados para os outros escravos.

Existem quatro variações para este protocolo, são elas:

- Modbus Plus: Baseado no meio físico RS-485. Utilizado para comunicar vários mestres e escravos, ou seja, CLPs entre si, módulos I/Os e interfaces homem - máquina. É a única variante que não é de domínio

publico. Possui vários recursos adicionais como roteamento, diagnósticos, endereçamento e consistência de dados.

- Modbus RTU: Baseado no meio físico RS-232 e RS-485. Utiliza o modelo mestre-escravo e permite a conexão de somente um mestre podendo ter vários escravos. Utilizado para comunicar CLPs entre si, módulos I/Os e interfaces homem - máquina. Transmite dados em formato binário de 8 *bits*. Isso possibilita a compactação dos dados em pequenos pacotes.
- Modbus ASCII: Baseado no meio físico RS-232 e RS-485. Utiliza o modelo mestre-escravo. Utilizado para comunicar CLPs entre si, módulos I/Os e interfaces homem - máquina. Utiliza caracteres ASCII para transmitir os dados, com isso consumindo mais recursos de rede.
- Modbus TCP/IP: Baseado no meio físico *Ethernet*. Utilizado para comunicação entre CLPs e Supervisório. Permite utilizar vários mestres e vários escravos. Aqui os dados são encapsulados em formato binário em frames TCP para utilização do meio físico *Ethernet*.

A comunicação em Modbus se dá em um *frame* que contém o endereço do escravo, o comando a ser executado, as informações propriamente passada e uma verificação de consistência de dados, o CRC (*Cyclic Redundancy Check*).

Os principais comandos Modbus são mostrados no Quadro 1 abaixo:

Código do comando	Descrição
1	Lê um número variável ¹ de saídas digitais (bobinas)
2	Lê um número variável ¹ de entradas digitais
3	Lê um número variável ¹ de registros retentivos (saídas analógicas ou memórias)
4	Lê um número variável ¹ de registros de entrada (entradas analógicas)
5	Força uma única bobina (altera o estado de uma saída digital)
6	Preset de um único registro (altera o estado de uma saída analógica)
7	Lê exceções ² (registros de erro)
8	Várias funções de diagnóstico
15	Força uma quantidade variável ¹ de bobinas (saídas digitais)
16	Preset de uma quantidade variável ¹ de registros (saídas analógicas)

Quadro 1 – Comandos Modbus

Fonte: Autoria própria

A verificação de consistência de dados é feita através do uso de um cálculo chamado CRC (*Cyclic Redundancy Check*). Ele é calculado e anexado ao fim da mensagem Modbus e, verificada no equipamento com que está comunicando.

O CRC possui as seguintes características:

- Como todos os bits são usados no cálculo, a mudança em apenas um *bit* provoca uma mudança no CRC;
- Mudanças pequenas nos dados levam a CRC muito diferentes.

Assim, usando o exemplo de uma determinada comunicação, onde o mestre precisa ler os 3 registros à partir do 108 (endereço 108 ao endereço 110) no escravo de endereço 17, a palavra de comunicação seria desta forma:

endereço	comando	end. dos registros		quant. de registros		CRC
11	03	00	6B	00	03	7E

Quadro 2 - Pergunta comunicação Modbus

Fonte: Autoria própria

E a resposta, simulando valores nos registros lidos, será a seguinte:

endereço	comando	num. bytes	valor dos registros					CRC	
11	03	06	02	2B	00	00	00	64	55

Quadro 3 - Resposta comunicação Modbus

Fonte: Autoria própria

1.5.8 Tecnologias de comunicação sem fio

Como visto anteriormente, a solução para a inviabilidade de uma comunicação via cabos é utilização de tecnologias sem-fio. Segundo Silva (2004) estas tecnologias utilizam, na maioria das vezes, radiofrequências ou raios infravermelhos. As informações são moduladas pelo emissor e são enviadas através de ondas eletromagnéticas numa certa frequência. O receptor, sintonizado na mesma frequência do emissor, recebe o que lhe diz respeito e rejeita as outras frequências. Exemplos de tecnologias de comunicação sem-fio e suas limitações são:

- IrDA (alcance pequeno, comunicação ponto-a-ponto);
- Bluetooth (baixo alcance);

- Wi-Fi (necessita estar ao alcance de um *hotspot*);
- GSM (alto custo, velocidade reduzida).

Outro exemplo de comunicação sem-fio, e que será usada neste trabalho, é a tecnologia GPRS.

1.5.9 GPRS

Segundo Sverzut (2005, cap. 9) o serviço GPRS (*General Packet Radio Service*) é uma evolução da tecnologia GSM (*Global System for Mobile Communication*) que permite que informações trafeguem utilizando esta infraestrutura de telefonia móvel.

A tecnologia GPRS permite o desenvolvimento de novos serviços, trazendo vários benefícios aos usuários de sistemas móveis celulares. Um desses benefícios é a mobilidade, que permite ao usuário manter uma comunicação de voz e dados durante o seu deslocamento entre áreas, outra é a conectividade. A combinação desses novos serviços gera uma grande variedade de aplicações, agregando valores aos usuários de sistemas móveis de celulares.

O GPRS é um serviço “sempre ativo”, diferente da tecnologia GSM que estabelece a conexão através da comutação de circuitos. Isto significa que o usuário só envia/recebe informações da central quando realmente necessário. Isto faz com que a capacidade da rede aumente e por consequência o número de usuários ligados à rede também. Permite o transporte de dados por pacote, oferecendo uma taxa de transferência de dados muito mais elevadas que as taxas de transferência das tecnologias anteriores, como CDMA, TDMA e GSM.

De acordo com Sverzut (2005, pág. 316), a comutação de pacotes possui as seguintes características:

- Tráfego intermitente e formado por rajadas (*bursts*): o fluxo de tráfego não é contínuo, porém com rajadas intermitentes.
- Tráfego assimétrico: o fluxo de tráfego de/para os usuários não é simétrico.

1.5.10 Padrão de comunicação OPC

Segundo Fonseca 2002, o padrão OPC (*Object Linking and Embedding (OLE) for Process Control*) surgiu como uma forma de padronizar regras de comunicação entre aplicações da plataforma Windows, ou seja, esse padrão define interfaces entre aplicações que estão no nível do usuário, acima da sétima camada do modelo OSI, camada na qual se encontram os *drivers* de comunicação.

Analisando a Figura 5, pode se observar que esse padrão define somente os métodos e tipos de dados que poderão ser utilizados entre a aplicação chamada de cliente e a aplicação chamada de servidor.

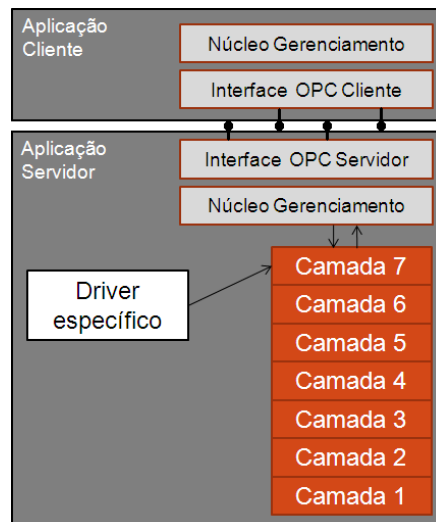


Figura 5 - Interfaces entre cliente e servidor segundo o padrão OPC

Fonte: Fonseca (2002)

Ainda segundo Fonseca (2002), o desempenho da comunicação OPC é semelhante ao desempenho apresentado por sistemas que utilizam *drivers* de comunicação específicos e otimizados. Normalmente, os *drivers* específicos possuem um ótimo desempenho após serem devidamente depurados e otimizados. Como um servidor OPC nada mais é do que uma camada de software a mais para implementar as interfaces padrões e os mecanismos de comunicação com o cliente, é de se esperar que o desempenho do mesmo só seja afetado em relação a comunicação com o cliente e não com o dispositivo de campo.

1.5.11 Sistema de supervisão (SCADA)

Boyer (2004) cita que os sistemas de supervisão SCADA (*Supervisory Control And Data Acquisition*) são sistemas que utilizam *softwares* onde o usuário pode monitorar, manipular e controlar variáveis de processos industriais disponibilizadas em tempo real, conectados através de *drivers* de comunicação. Assim, é possível visualizar o estado real do sistema através de um conjunto de gráficos e relatórios possibilitando que decisões sejam tomadas tanto por parte do operador quanto por parte do próprio *software* automaticamente.

Esses sistemas também são utilizados para gerir as empresas, pois através dos dados obtidos em tela, é possível medir no próprio programa o desempenho de uma linha de produção e identificar onde ou o quê está causando um atraso no processo.

De modo geral, a maioria dos sistemas supervisórios permite a comunicação com inúmeros protocolos e equipamentos, podendo acomodar tanto sistemas locais quanto geograficamente distribuídos.

Com o objetivo de aumentar a flexibilidade, confiabilidade e conectividade, os sistemas SCADA possuem varias ferramentas que são divididas em módulos e blocos, estes blocos se comunicam entre si com a finalidade de disponibilizar as informações na tela gráfica, possibilitando ao usuário tirar as conclusões necessárias para o controle seguro do processo. Alguns dos módulos são:

- Núcleo de processamento;
- Gerenciamento de Alarmes;
- Interface gráfica;
- Relatórios;
- Gráficos.

Os sistemas SCADA podem coletar dados do processo e armazená-los em banco de dados, e estes dados podem ser integrados a sistemas MES (*Manufacturing Execution Systems*), ERP (*Enterprise Resource Planning*), entre outros. E podem também fornecer dados em tempo real para sistemas que realizam cálculos OEE (*Overall Equipment Effectiveness*), sistemas SFC (*Shop Floor Control*).

1.5.12 CLP

O Controlador Lógico Programável, ou simplesmente CLP, tem revolucionado os comandos e controles industriais desde seu surgimento na década de 70.

Antes do surgimento dos CLPs as tarefas de comando e controle de máquinas e processos indústrias eram feitas por relés eletromagnéticos, especialmente projetados para este fim, e devido a grande dificuldade de mudar a lógica de controle dos painéis de comando a cada mudança na linha de montagem viu-se a necessidade de criar um equipamento versátil e de fácil utilização, chamado inicialmente de MODICON (*Modular Digital Controller*)

O primeiro CLP surgiu na indústria automobilística, mais precisamente na divisão de transmissão automática da *General Motors*, até então um usuário frequente dos relés eletromagnéticos. . Eram utilizados para controlar operações sequenciadas e repetitivas numa linha de montagem. A primeira geração de CLPs utilizou componentes discretos como transistores e CIs com baixa escala de integração.

Este equipamento foi batizado nos Estados Unidos como PLC (*Programmable Logic Controller*), em português CLP (Controlador Lógico Programável) e este termo é registrado pela Allen Bradley (fabricante de CLPs).

É um equipamento eletrônico digital com *hardware* e *software* compatíveis com aplicações industriais (ABNT).

Os controladores lógicos programáveis (CLPs) são equipamentos eletrônicos de última geração, utilizados em sistemas de automação flexível. Permitem desenvolver e alterar facilmente a lógica para acionamento das saídas em função das entradas. Desta forma, pode-se utilizar inúmeros pontos de entrada de sinal para controlar pontos de saída de sinal (cargas).

As vantagens dos controladores lógicos programáveis em relação aos sistemas convencionais são:

- Ocupam menos espaço;
- Requerem menor potência elétrica;
- Podem ser reutilizados;
- São programáveis, permitindo alterar os parâmetros de controle;

- Têm maior confiabilidade;
- Sua manutenção é mais fácil;
- Oferecem maior flexibilidade;
- Permitem interface de comunicação com outros CLPs e computadores de controle;
- Permitem maior rapidez na elaboração do projeto do sistema.

Desde seu aparecimento, muita coisa evoluiu nos controladores lógicos programáveis. Esta evolução está ligada diretamente ao desenvolvimento tecnológico da informática, tanto em termos de *software* quanto de *hardware*.

Pode-se dizer que o CLP é um “microcomputador” aplicado ao controle de um sistema ou de um processo, composto de módulos de entradas digitais ou analógicas.

A lógica a que são submetidas às entradas para gerar as saídas é programada pelo usuário do sistema.

As saídas também podem ser digitais ou analógicas. A exemplo das entradas, as saídas digitais são tratadas em conjuntos de 8 ou 16 e as analógicas são resultado da conversão de um valor digital gerado pela UCP.

O CLP possui uma Unidade Central de Processamento (UCP), que é responsável pelo processamento do programa, isto é, coleta os dados dos cartões de entrada, efetua o processamento segundo o programa do usuário, armazenado na memória, e envia o sinal para os cartões de saída como resposta ao processamento.

2. DESENVOLVIMENTO

2.1 ARQUITETURA DO SISTEMA

O modelo de arquitetura deste projeto é fundamentado na teoria de sistemas distribuídos, ou seja, seguirá padrões que melhoram o seu desempenho através do posicionamento e relacionamento entre os elementos deste sistema, atendendo as necessidades atuais e garantindo futuramente uma expansão segura. O objetivo de se pesquisar um modelo de arquitetura para o sistema é fazer com que o sistema seja confiável, adaptável, gerenciável e escalável. O sistema, se não for corretamente planejado, pode, a partir de uma falha em um nó da rede causar a não entrega dos dados aos outros nós.

Neste projeto, será utilizada uma arquitetura cliente-servidor, onde o “Servidor de Dados GPRS – Modbus TCP” será **cliente** para o CLP (servidor) e **servidor** para o Sistema de Supervisão (cliente).

Como pode ser visto na Figura 6, será utilizado neste projeto um CLP Altus FBs, um CLP de pequeno porte suficiente para o controle de alguns equipamentos em campo e para a comunicação através de sua porta serial RS-232 com um modem GPRS DTU, que é necessário para se estabelecer a comunicação sem-fio através do serviço de dados móvel GPRS.

A operadora de telefonia móvel disponibilizará as informações enviadas pelo modem através da *Internet* para um computador com IP (fixo preferencialmente), que servirá como servidor de dados para uma rede local. Este então enviará as informações via *Ethernet* para um Sistema de Supervisão. Todos os meios e equipamentos utilizados no projeto que foram descritos nos parágrafos acima, são mostrados pela Figura 7.

Apesar de os dados fluírem pela rede de telefonia, não é possível que dados de terceiros interfiram no sistema, pois a conexão é feita através de uma determinada porta *Ethernet*.

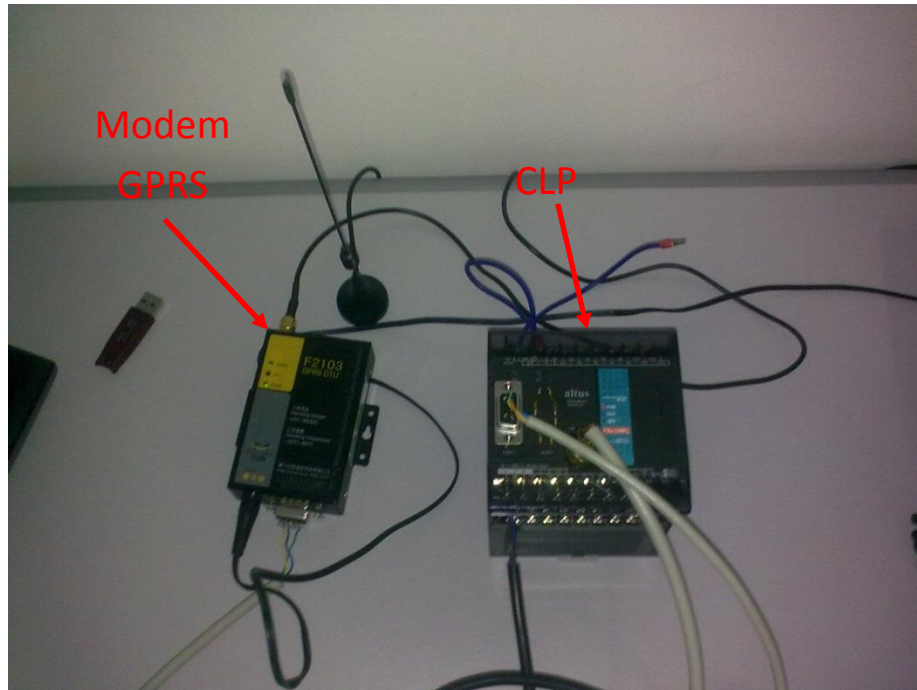


Figura 6 - Bancada de teste com o CLP conectado ao modem GPRS
 Fonte: A autoria própria

Foi desenvolvido para o “Servidor de Dados GPRS – Modbus TCP” um *driver* Modbus, ou seja, foram feitos programas para que se transfiram os dados de acordo com esse protocolo.



Figura 7 - Topologia do projeto
 Fonte: A autoria própria

Seguindo a lógica do sistema, o “Servidor de Dados GPRS – Modbus TCP” deve atualizar suas bases de dados (envio e recebimento de dados) obedecendo a um ciclo de tempo pré-determinado, configurado no *software*. A Figura 8 ilustra a

arquitetura em que o Servidor de Dados Local é o mestre da comunicação para o CLP, e o início dessa troca de informação deve ser uma iniciativa dele. Sendo assim, o “Servidor de Dados GPRS – Modbus TCP” deve fazer uma conexão no endereço IP e porta especificado no modem GPRS.

Atualmente as operadoras de telefonia móvel não fornecem mais um endereço IP fixo, sendo necessário a utilização de outros meios de conexão, que no caso deste trabalho, se dá através de um DDNS (DNS (*Domain Name System*) Dinâmico).



Figura 8 - Relação Cliente-Servidor e Mestre-Escravo
Fonte: Autoria própria

2.2 COMUNICAÇÃO ENTRE OS DISPOSITIVOS

Primeiramente um estudo de custo/benefício foi realizado para que fosse escolhido qual entre os protocolos TCP e UDP deveria ser utilizado junto ao IP na comunicação GPRS. Verificando que o sistema deve prezar pela garantia de entrega a cada envio de dado, chegou-se a conclusão que o protocolo TCP seria o mais adequado, pois apesar de custar um pouco mais caro, é muito mais confiável devido existência da garantia de entrega (ACK). Já existem planos de acesso à *internet* pelo celular onde se paga quando houver conexão, o valor de R\$0,50 por dia. Isso dá um valor mensal de R\$15,00! Porém, para realizar um cálculo de custo da transmissão, pegaremos outra operadora, que possui planos de 10MB por R\$9,90 e o que exceder fica por R\$0,55 o MB adicional (**Erro! Fonte de referência não encontrada.**).

Tabela 1 - Comparação de custo TCP versus UDP

	TCP	UDP
Bytes cabeçalho / Socket (Envio e receb.)	340	84
Bytes de dados	160	160
Bytes por Transmissão (Total por conexão)	500	244
Custo por Byte*	R\$ 0,00000055	R\$ 0,00000055
Número de conexões por dia	666	1366
Mensalidade*	R\$ 9,90	R\$ 9,90
TOTAL MENSAL	R\$ 9,90	R\$ 9,90

Fonte: Aatoria própria

Notas:

(1) Dados segundo operadora.

Como o total de transmissão para cada conexão TCP fica em 500 *bytes*, para se realizar o consumo de 10MB, seriam necessárias 20.000 conexões por mês, o que dá 666 conexões por dia. Ou seja, uma conexão a cada 2,5 minutos e ainda sobram dados para consumir.

Segundo o estudo de custo/benefício, chegou-se à conclusão que os protocolos que serão utilizados na comunicação entre os dispositivos são:

- Modbus RTU (*Remote Terminal Unit*);
- Modbus TCP;
- TCP/IP.

Como mostra a Figura 9, o CLP deve enviar os *frames* de dados à operadora de telefonia móvel através do protocolo TCP/IP. Dentro deste protocolo, porém, os dados serão encapsulados pelo próprio CLP em protocolo Modbus. Ao chegar à operadora, esse mesmo pacote de dados é retransmitido via *Internet* para um o servidor local, que possui preferencialmente um IP fixo. Este tratará as informações recebidas e disponibilizará os dados para a o sistema supervisório através do protocolo Modbus TCP.

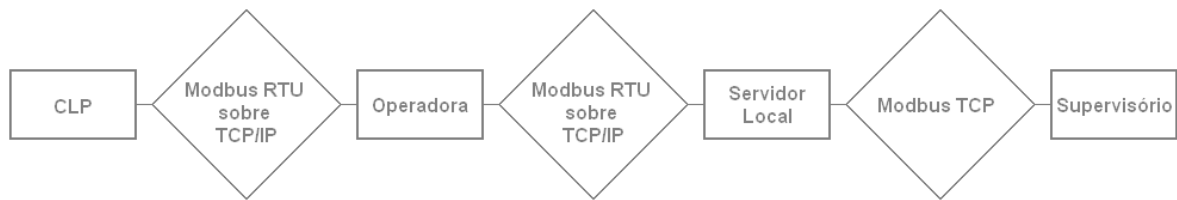


Figura 9 - Comunicação entre os nós da rede
Fonte: Autoria própria

2.3 COMUNICAÇÃO GPRS

Com a intenção de conectar o *modem* ao serviço GPRS, é necessário que o *modem* execute uma série de comandos AT para configurá-lo. Estes comandos verificam se o *modem* está preparado para a conexão, indicam a porta e o endereço IP do servidor local que receberá os dados, o seu tipo de funcionamento. Após a conexão ser estabelecida inicia-se a troca de dados. Caso alguma das etapas falhar, o *modem* reinicia seu processo de configuração.

O *modem* funcionará como cliente *Telnet*, dado que o “Servidor de Dados GPRS – Modbus TCP” implementa a função de servidor *Telnet*. O servidor abre uma determinada porta TCP e fica “escutando”, até que alguém se conecte.

2.4 SERVIDOR MODBUS TCP

Com a conexão nesta porta estabelecida, o “Servidor de Dados GPRS – Modbus TCP” passa a ser um cliente do CLP, enviando/recebendo informações para o/do processo.

No *software*, as portas de comunicação com a *Internet* são configuradas em um arquivo chamado “CfgApp.ini” e com o SCADA podem ser configuradas como indicado na Figura 10 e Figura 11.

```

CfgApp.ini - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
[Config]
TCPPortIniServers=9000
TCPPortIniRemotas=10000
  
```

Figura 10 - Configurando portas TCP
Fonte: Autoria própria

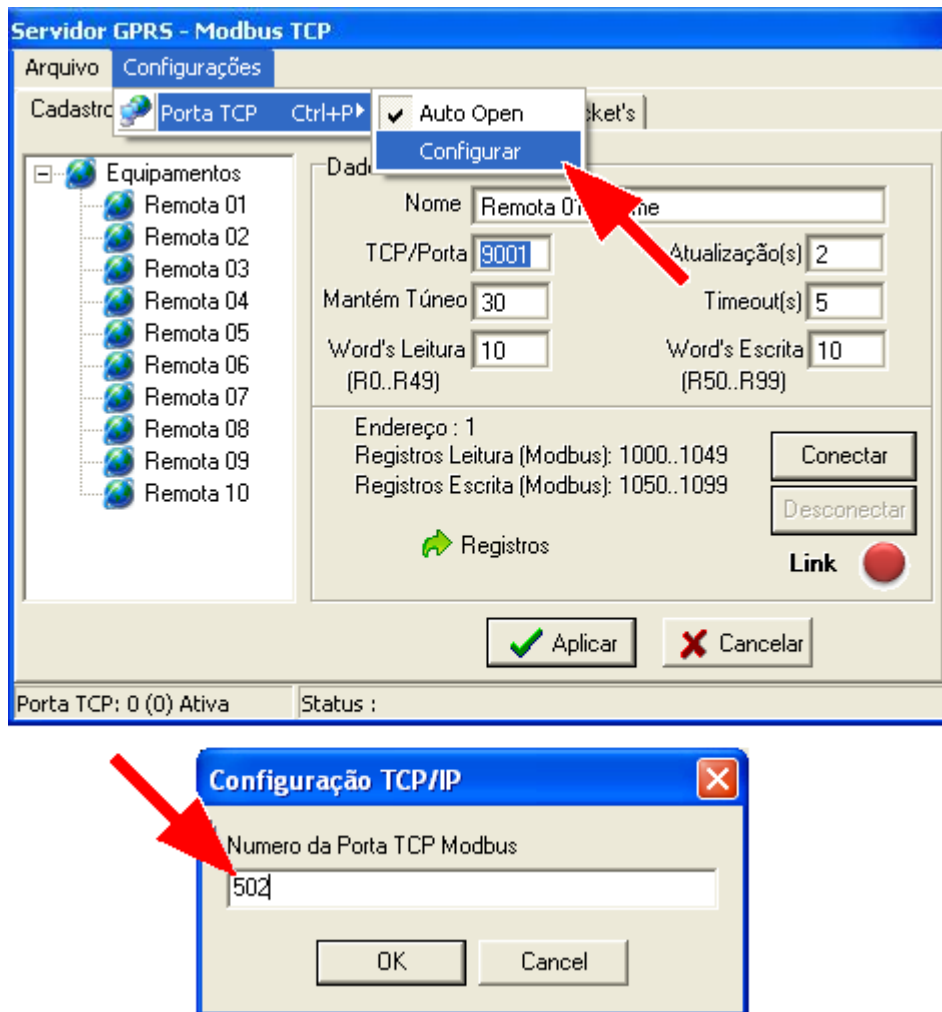


Figura 11 - Configurando portas Modbus TCP
Fonte: Autoria própria

Ao receber esses dados, o “Servidor de Dados GPRS – Modbus TCP” verifica o CRC do frame respondido pelo CLP e caso esteja correto, disponibiliza para o SCADA via Modbus TCP e indica os valores na tela do *software* “Servidor de Dados GPRS – Modbus TCP”. A Figura 12 mostra a tela de exibição dos valores que podem ser alterados clicando duas vezes em cima da célula desejada.

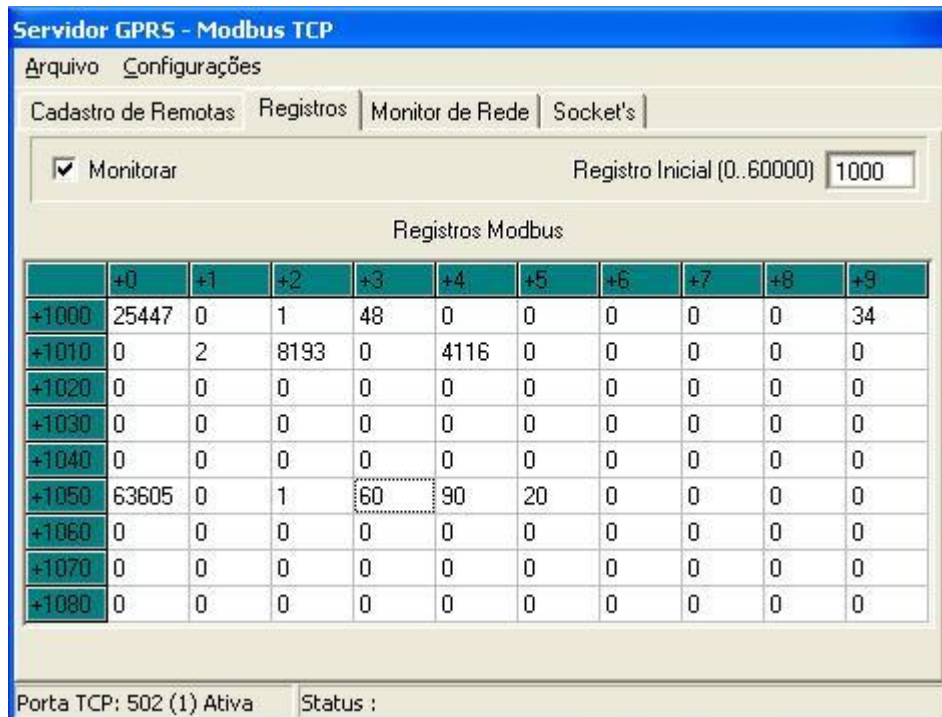


Figura 12 - Tela do aplicativo que controla o protocolo Modbus (Servidor)
Fonte: Autoria própria

Para estabelecer a conexão entre o servidor local e o SCADA, será utilizado um OPC (*OLE for Process Control*) Server. Este servidor/cliente define interfaces entre aplicações que estão no nível do usuário, acima da sétima camada do modelo OSI, camada na qual são encontrados os *drivers* de comunicação, ou seja, tem como função principal criar um elo de comunicação entre qualquer *driver* de comunicação e um sistema supervisório.

2.5 TELA DO SISTEMA SUPERVISÓRIO

Foi utilizado no projeto o programa de supervisão iFix Dynamics, que é responsável pela parte visual e controle operacional do sistema. É através dessa interface que o usuário final visualiza os valores obtidos no campo e faz as alterações necessárias em nível de controle.

Visualiza-se na Figura 13, um tanque, uma bomba, um medidor de nível e dois *setpoints* de operação da bomba, um de nível alto, para ligar a bomba, e outro de nível baixo, para desligar a bomba e as residências que serão abastecidas pela água recalçada pela bomba.

Estes *setpoints* atuam no sistema quando este está operando em modo automático. Quando em modo manual, o operador poderá ligar e desligar a bomba através dos comandos de operação.

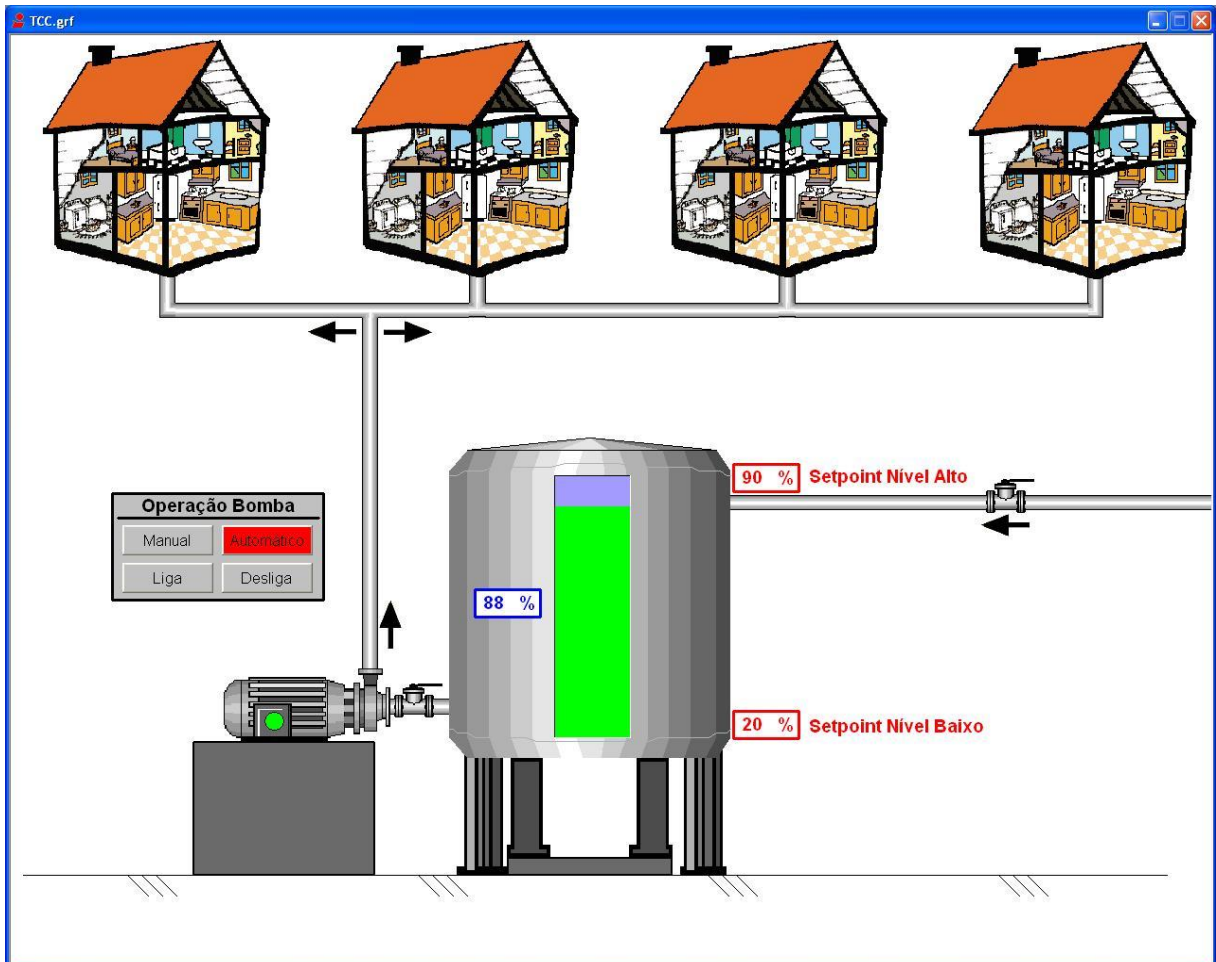


Figura 13 - Equipamentos de campo
Fonte: Autoria própria

O seguinte dado será medido em campo:

- Nível do tanque;

Será controlado:

- Acionamento remoto da bomba.

2.6 TESTES

Com o objetivo de provar a eficiência do sistema, foram realizados alguns testes práticos para a validação. Duas partes do projeto foram testadas separadamente para que, caso houvesse algum erro, fosse facilmente encontrado.

Inicialmente foi testada a comunicação entre o servidor local e o supervisor, para isso foram atribuídos alguns valores às variáveis do sistema para provar o envio/recebimento destas.

As telas ilustradas na Figura 14 mostram a eficiência dessa comunicação via Modbus TCP através do OPC, pois como podem ser visualizados, os valores se repetem nas telas e a qualidade de sinal está “Good”, que representa boa qualidade.

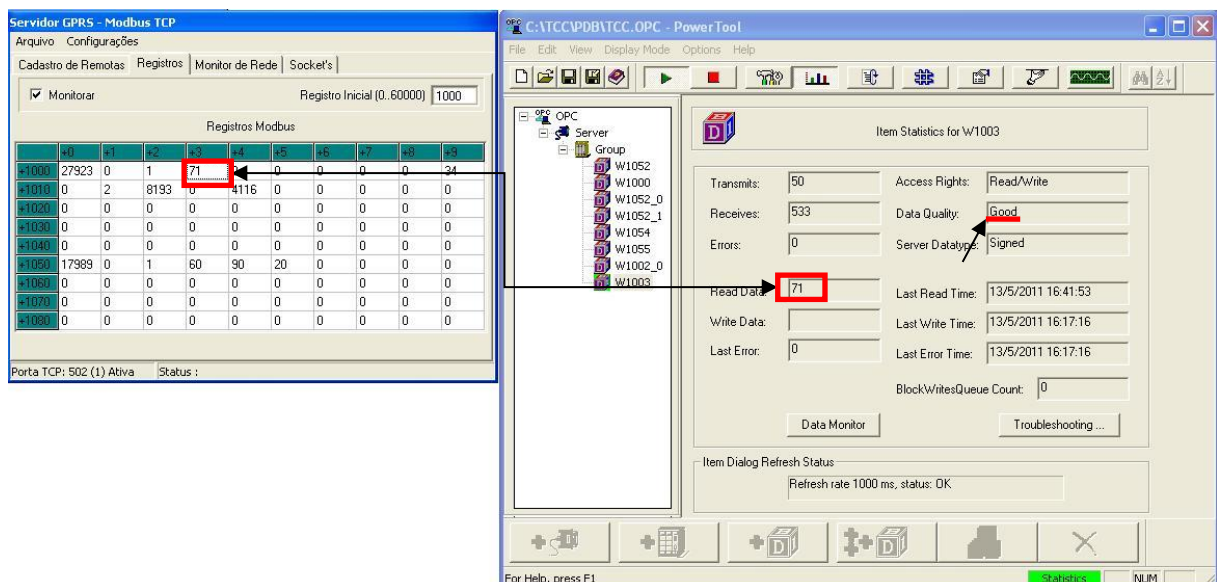


Figura 14 - Comunicação entre o servidor Modbus e o supervisor
Fonte: Autoria própria

A segunda parte de testes foi realizada para comprovar a funcionalidade do sistema como um todo, ou seja, desde o CLP até o supervisor. Os dispositivos de campo foram simulados através de uma lógica randômica programada no CLP para que valores fossem alterados e pudessem comprovar o envio dos dados. De acordo com a Figura 15, pode-se observar o estabelecimento das duas conexões entre as três partes do sistema, CLP, servidor Modbus e o supervisor. As portas utilizadas são as 9001 (Servidor GPRS) e 10001 (Servidor Modbus TCP) para a conexão via *Internet* e a 502 para a conexão com o supervisor.

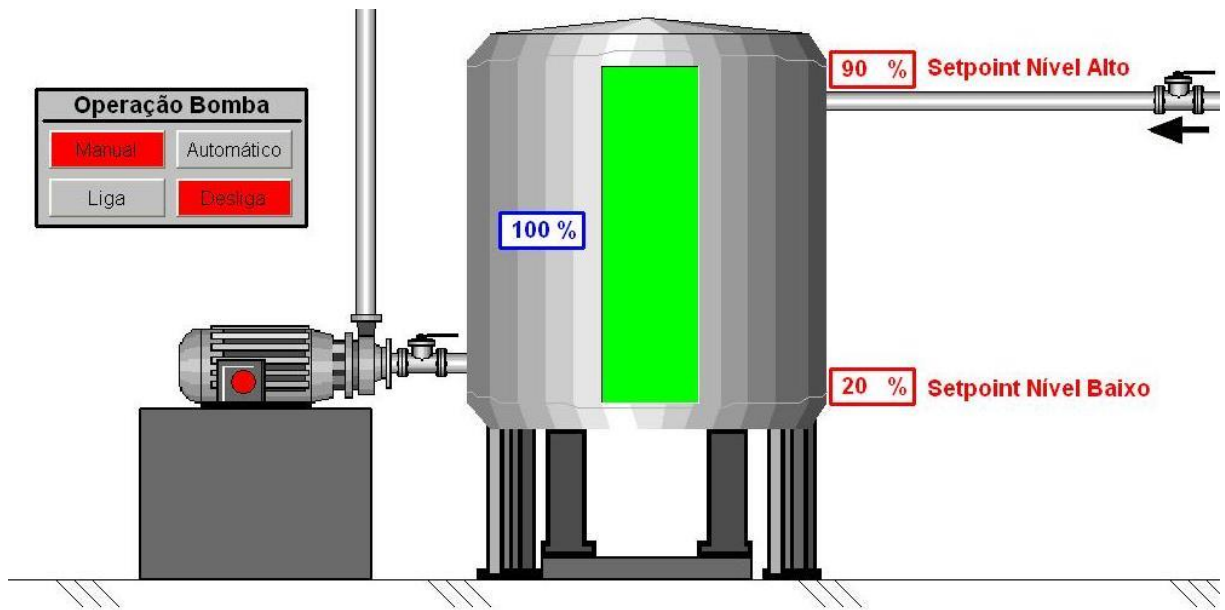


Figura 16 - Supervisório comunicando com servidor Modbus
Fonte: Autoria própria

3. CONCLUSÃO

A comunicação de dados no ambiente industrial é uma necessidade clara. Tendo em vista a eventual impossibilidade de transferência de dados via cabo por qualquer que seja o motivo, a utilização de uma rede sem-fio torna-se mandatória.

Este projeto mostrou a viabilidade da implementação de um sistema controle e a integração de vários assuntos relacionados à monitoração remota e controle de dispositivos em campo através da comunicação sem-fio. Tal abordagem foi realizada com o objetivo de obter conhecimento suficiente para tornar esse sistema o mais confiável e funcional possível.

O projeto demonstrou eficiência no quesito arquitetura, pois o sistema se mostrou pronto para receber inúmeras unidades remotas semelhantes às encontradas em campo sem que seu desempenho fosse alterado. Utilizando o CLP FBs da Altus obteve-se um controle limitado de dispositivos em campo devido à sua capacidade. Caso seja necessário, a implementação de uma lógica de controle mais complexa, deve-se proceder à substituição desse CLP por um de maior porte, como por exemplo, um CLP da série Ponto da Altus.

Os protocolos utilizados mostraram-se adequados no transporte das informações, mas, provou-se que na troca de TCP/IP para UDP/IP o custo pela transmissão pode ser reduzido, conforme mostrado na Tabela 2, onde a quantidade de dados trafegados na mensagem UDP/IP é menor do que a TCP/IP. Neste caso, porém, a perda de confiabilidade é muito grande. Devem ser avaliados planos de dados GPRS em que é cobrada somente uma mensalidade. Não há nestes planos, custo adicional por volume de dados, podendo ser aumentando a quantidade de acessos à rede GPRS.

O serviço GPRS trouxe uma solução simples e eficaz para o projeto, apresentando confiabilidade e taxas de transferências suficientes para a aplicação. A grande questão de se utilizar essa tecnologia é quanto a sua disponibilidade e variação de velocidade, pois, depende plenamente da cobertura GSM oferecida pelas companhias de telefonia móvel. Outra informação importante é sobre a velocidade mínima contratada, que no Brasil é de 10% da nominal. Isso significa que se contratar uma velocidade de *internet* de 1MB, a operadora pode te entregar 100KB de velocidade.

REFERÊNCIAS

BOYER, Stuart A. **SCADA: supervisory control and data acquisition**. 3.ed. Research Triangle Park : ISA, 2004. Cap. 2.

COULORIS, G. F.; DOLLIMORE, J.; KINDBERG, T. **Sistemas distribuídos: conceitos e projetos**. 4.ed. Porto Alegre: Bookman, 2007. Cap. 3 e Cap. 4.

Especificação do protocolo Modbus. Disponível no site < http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf >. Acesso em: 2 de março de 2011.

FARIAS, Paulo Cesar Bento **Redes - Básico**. Disponível no site < <http://www.juliobattisti.com.br/tutoriais/paulocfarias/redeswireless024.asp> >. Acesso em: 26 de novembro de 2011

FONSECA, Marcos de Oliveira. OPC – Uma abordagem prática. Disponível no site < <http://www.cpdee.ufmg.br/~seixas/PaginaSDA/Download/DownloadFiles/OPCMarcosFonseca.PDF> >. Acesso em: 15 de abril de 2011.

Modbus Organization < <http://www.modbus.org/> >. Acesso em: 10 de fevereiro de 2011.

OPC Foundation < <http://www.opcfoundation.org/> >. Acesso em: 17 de março de 2011.

SILVA, Adailton J. S. **As Tecnologias de Redes Wireless**. Disponível no site < <http://www.rnp.br/newsgen/9805/wireless.html> >. Acesso em: 03 de dezembro de 2010

SIMOMURA, Bruno Celio **Sistemas Distribuídos**. Disponível no site < <http://www.artigonal.com/ti-artigos/sistemas-distribuidos-991878.html> >. Acesso em: 14 de dezembro de 2011

SVZERZUT, José Umberto. **Redes GSM, GPRS, EDGE e UMTS: Evolução a Caminho da Terceira Geração (3G)**. 1.ed. São Paulo: Érika, 2005. Cap. 9.

TANENBAUM, Andrew S.; STEEN, Martin Van. **Sistemas distribuídos: princípios e paradigmas**. 2.ed. São Paulo: Pearson Prentice Hall, 2007. Cap. 2.

4. ANEXOS

ANEXO A - Catálogo dos CLP's Altus:

http://www.altus.com.br/ftp/Public/Portugues/Catalogos%20de%20Produtos/Catalogo_CPs.pdf

ANEXO B - Catálogo da solução OPC utilizada no projeto.

http://www.abmicro.com.pl/appli/pdf_1/aiModbusTCP.pdf