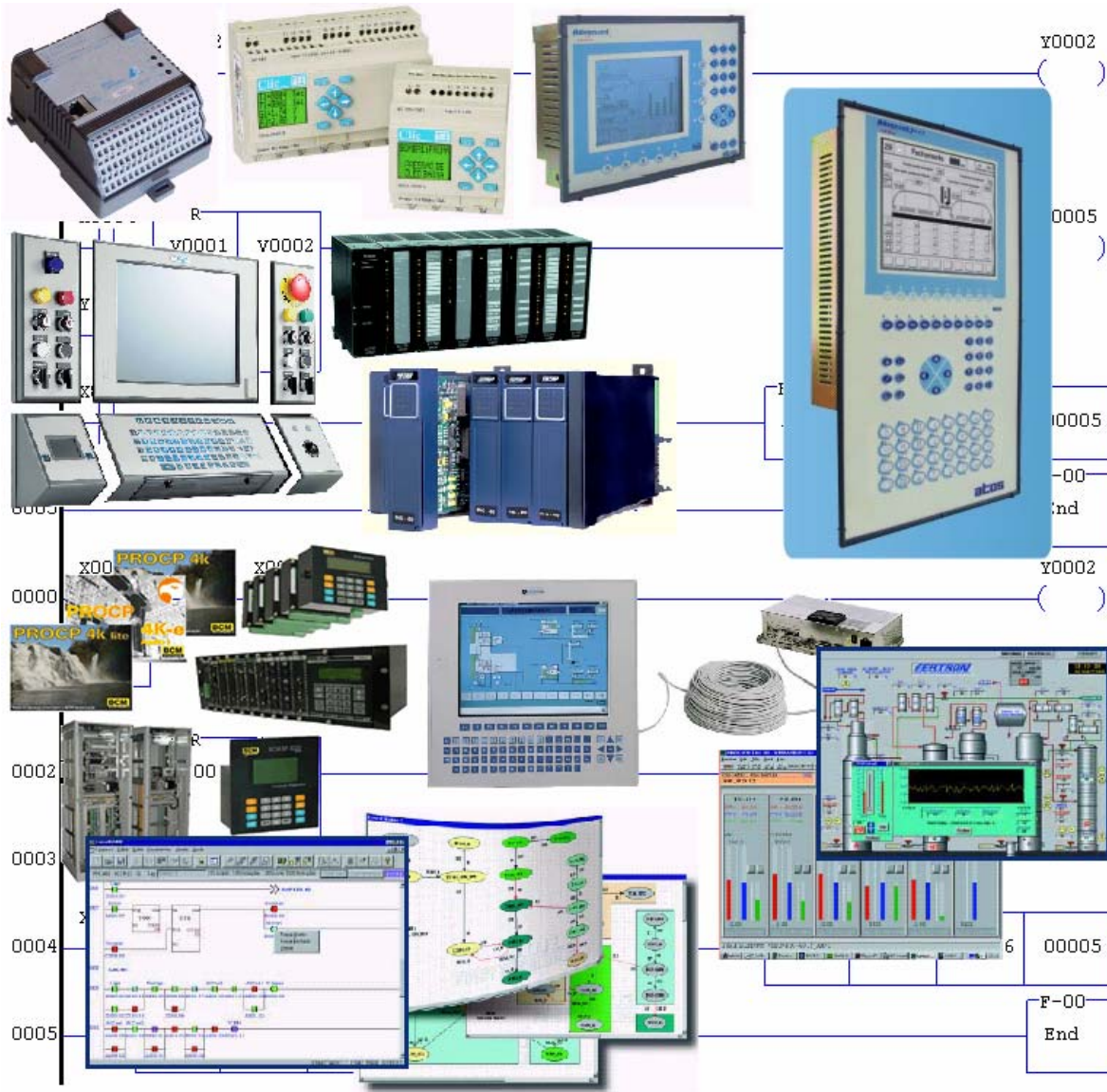


INFORMÁTICA INDUSTRIAL

Computador Industrial e Sistemas SCADA



1. Computadores para Controle de Processos

Os primeiros computadores foram desenvolvidos nos anos 50, os quais eram muito grandes e de custo elevado. Estes computadores foram utilizados principalmente em tarefas administrativas como folha salarial, contabilidade e operações bancárias. As operações desempenhadas eram normalmente processos por batelada.

Os microprocessadores foram desenvolvidos nos anos 70, iniciando uma dramática revolução que resultou em computadores menores e mais baratos. Nos anos 70, muitos sistemas de controle foram desenvolvidos utilizando microprocessadores como controladores.

A maior vantagem dos computadores em comparação com a lógica por interligação elétrica, é que a função de controle é programada, podendo ser modificada facilmente. Computadores também podem facilmente manipular grandes volumes de dados e executar inúmeros cálculos. Um computador padrão, no entanto, não é preparado para comunicar com equipamentos industriais. Outra desvantagem é o nível de conhecimento necessário para o desenvolvimento de programas. Os primeiros sistemas de controle baseados em computadores necessitam de interfaces especiais para manipular os sinais provenientes dos transdutores e atuadores. Estas interfaces normalmente tinham de ser desenvolvidas especialmente para cada planta industrial. Desde então, diversos fabricantes desenvolvem módulos de interfaces padrões para sinais digitais e analógicos.

1.1 Histórico no Brasil

A atividade de pesquisa em computação começou a ser realizada institucionalmente a partir dos anos 70. Vocações por especialidades na área começaram a surgir. Os novos departamentos foram criados mais próximos da Engenharia Elétrica ou da Matemática. Por exemplo, em 72 o Departamento de Engenharia Elétrica da USP projetou o minicomputador "Patinho Feio" e nos dois anos subsequentes a USP junto com a PUC-Rio desenvolveram, respectivamente, o hardware e o software básico do G-10 (primeiro computador industrial produzido no Brasil).

1.2 Métodos de programação

Todo programa de computador consiste em um número de instruções as quais dizem para o computador o que é para ser feito quando o programa é executado. Como os computadores entendem apenas informações binárias, as instruções dos computadores são muito diferentes da linguagem verbal utilizada por nós para descrever as ações que queremos que sejam executadas. Em programação, entretanto, vários artifícios são empregados para transformar nossa descrição verbal na linguagem própria dos computadores. Estes artifícios são implementados em linguagens de programação, as quais podem ser adquiridas a um custo bem acessível.

1.3 Código de máquina e assembler

Muitos computadores têm um limitado conjunto de instruções que desempenham operações simples tais como coletar um dado, armazenar o dado, somar números, etc.

Através da combinação de um grande número destes códigos de máquina em extensos programas, o programador pode fazer com que o computador desempenhe funções complexas. Contudo, para que o programa possa funcionar, é muito importante que sejam seguidas as regras de como as instruções devem ser usadas e combinadas, normalmente chamado de sintaxe do programa.

Como os códigos de máquinas são binários ou hexadecimais, a tarefa de programação é simplificada através do uso de instruções assembler. Cada uma destas instruções tem um nome composto por três letras (código mnemônico), tal como LDA, para coleta de dados e ADD para soma de dois números. Os programas assembler são escritos através de um editor e, antes que os mesmos sejam executados, os códigos de máquina em hexadecimal. Esta tradução é feita por um outro programa chamado de assembler. Além da tradução do programa, o assembler também verifica a sintaxe dos programas e calcula os saltos lógicos dentro de um programa. A assemblagem, processo realizado pelo assembler, de um programa é feita para um determinado tipo de computador ou sistema operacional, sendo que existem programas capazes de fazer a assemblagem para outros sistemas operacionais.

Algumas desvantagens da programação em assembler consistem basicamente em:

- Faz-se necessário o conhecimento do funcionamento de computadores para facilitar a programação;
- O programa final é destinado a um tipo de computador ou sistema operacional, não sendo facilmente portátil para os outros;
- O problema deve ser devidamente estruturado para facilitar a utilização do conjunto de instruções de um determinado computador.

Apesar destes pontos contrários, a linguagem assembler permite um ótimo desempenho e uma melhor utilização da memória do computador. Estes fatores podem ser determinantes em determinados sistemas de controle. O assembler é chamado de linguagem de baixo nível, em função do mesmo ser próximo à forma como os computadores processam os dados.

Existem programas chamados de depuradores (debuggers), os quais permitem executar o programa passo-a-passo, de forma que possam ser simulados os dados do sistema real sem a necessidade de conexão com a planta industrial, facilitando a depuração de erros.

1.4 Compilação e Interpretação

A programação é facilitada significativamente quando a mesma é feita utilizando-se uma linguagem de alto nível, a qual é traduzida para o código de máquina através de um programa chamado de interpretador ou compilador.

A diferença entre compiladores e interpretadores é que o compilador primeiramente traduz todo o programa para o código de máquina antes do mesmo ser executado, enquanto o interpretador traduz o programa instrução por instrução durante a execução do programa. Isto significa que os programas compiladores têm um processamento mais rápido. As linguagens de programação de alto nível mais comuns atualmente são o Pascal e a Linguagem C. Ambas são linguagens compiladas. Um exemplo de linguagem interpretada era o Basic mais antigo.

As instruções de uma linguagem de alto-nível são semelhantes às funções matemáticas, sendo, portanto mais intuitivas e fáceis de usar. Todas as linguagens de alto-

nível são altamente padronizadas, onde as partes principais de um programa podem ser escritas independentemente do tipo de computador onde ela será executada. A compatibilização com o tipo de computador é feita pelo programa compilador ou interpretador durante a tradução para o código de máquina. Programas escritos em linguagens de alto nível são chamados de programas fonte, enquanto o resultado da compilação é chamado de código objeto.

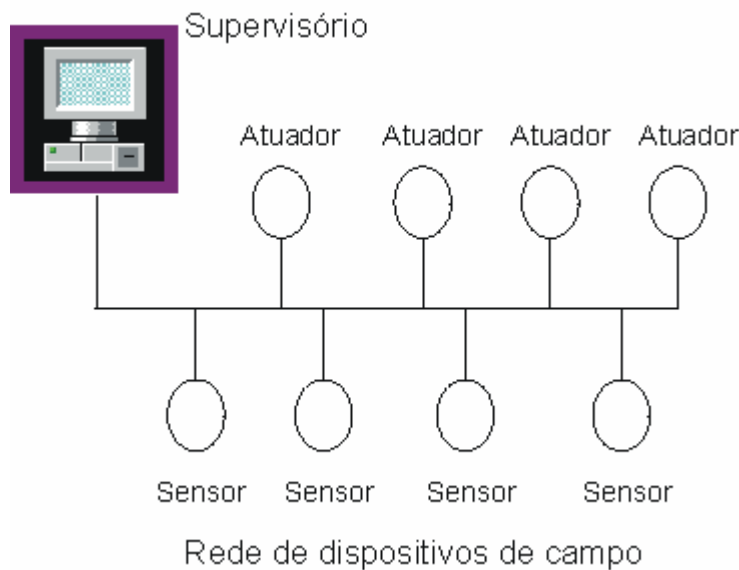
O programador não necessita conhecer detalhes do tipo de computador ou memória para a programação em uma linguagem de alto nível. Outra vantagem é que o programa gerado pode ser portado para outro tipo de computador, desde que exista o compilador adequado.

As desvantagens dos programas escritos em linguagens de alto nível são o maior consumo de memória e desempenho inferior em relação à linguagem assembler.

1.5 Redes de Campo

Os barramentos de campo ou Fieldbus, sistemas de troca de dados serial, constituem atualmente tecnologia de ponta para as mais diversas áreas de automação. De maneira simples, essa tecnologia se baseia na conexão dos elementos que compõe um sistema de automação através de um cabeamento comum, formando uma rede de dispositivos, que podem ser acessados individualmente, utilizando mensagens padronizadas por um protocolo. Com essa filosofia pode ser conseguida uma grande redução no trabalho de projeto, versatilidade na especificação de topologias de ligação (possibilitando redundância, por exemplo), facilidade de instalação física do sistema, entre outras. Isso gera, conseqüentemente, a diminuição dos custos e viabiliza mais facilmente as implantações. Pode-se então imaginar que as vantagens dos sistemas Fieldbus são ilimitadas mas atualmente um grande problema que se enfrenta é a existência de inúmeros protocolos concorrentes. Fieldbus Foundation, CEBus, WorldFIP, BitBus, ProfiBUS, Lonworks, BACNet, X10, EIB, CAB, CAN, etc... o projetista ou usuário se encontra num momento onde, com razão, tem-se o receio de fazer investimentos mal sucedidos por conta da não existência de um padrão de fato para as diversas áreas da automação. Por outro lado, não é possível esperar por essa padronização global, as tecnologias existem e precisam ser usadas. Nota-se, inclusive, que nem mesmo as soluções tradicionais, como as baseadas em PCL, atingiram tal grau de padronização. Cabe ao interessado na automação, analisar as diversas opções e procurar encontrar aquela que atende o maior número de necessidades do seu sistema.

Também não se espera que um único protocolo seja tão abrangente que supra todas as necessidades de todas as áreas da automação e do controle, mas que existam formas de integração entre uma e outra rede a fim de que dados sejam trocados entre elas. No entanto, alguns fabricantes consideram seus protocolos proprietários de interesse para seus negócios e não produzem conversores de protocolos para outros padrões e nem disponibilizam a documentação necessária para que terceiros o façam, mantendo os usuários de seus produtos isolados em suas plataformas.

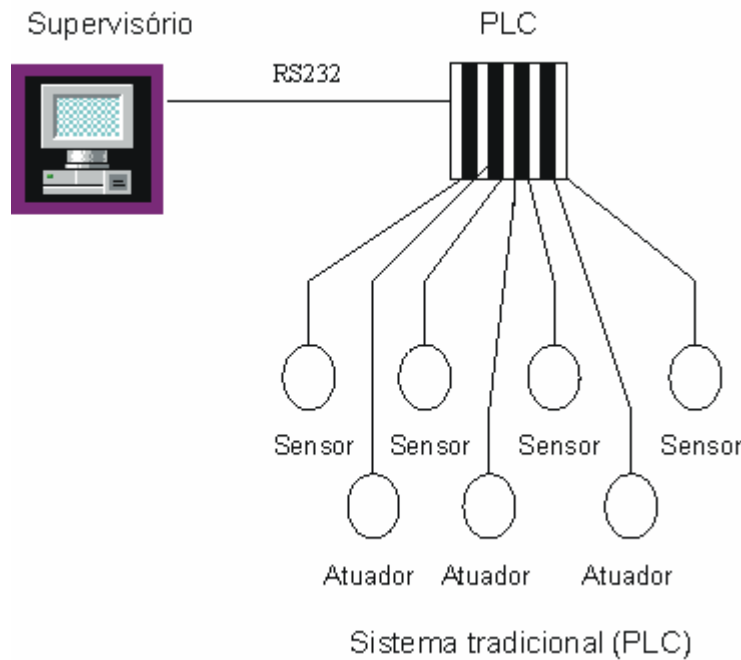


1.6 Controle Distribuído

Com o advento dos microprocessadores e microcontroladores, a eletrônica inteligente se tornou muito mais acessível e barata. Atualmente os sistemas de controle tendem a se tornar complexos, com um grande número de variáveis, ações e intertravamentos. Conseqüentemente, um controle centralizado se torna exageradamente complexo, caro e, possivelmente, lento. Dividir o problema em partes menores que podem ser resolvidas individualmente passa a ser então uma solução atrativa por muitos motivos: maior facilidade de desenvolvimento, operação e administração, confiabilidade do sistema como um todo (o mal funcionamento de uma parte não implica, necessariamente, no mal funcionamento de todo o sistema), manutenção simplificada, entre outros. A filosofia SDCD (Sistemas Digitais de Controle Distribuído), já é utilizada desde a década de 70 com bastante sucesso, tendo como base a utilização de terminais remotas conectadas aos dispositivos no campo e conectadas entre si a uma via de dados que por sua vez contém um elemento centralizador, que pode ser um CLP, um PC ou outro equipamento dedicado.

Com a entrada do conceito de Fieldbus no mercado, o controle distribuído ganha uma nova alternativa : a utilização de dispositivos inteligentes. Esses dispositivos são dotados de alguma capacidade de processamento, o que, aliado à conexão dos dispositivos em rede (Fieldbus), pode levar a uma solução onde os próprios dispositivos troquem mensagens entre si e o controle do sistema de automação seja de responsabilidade da rede de dispositivos e não mais de um elemento centralizador (CLP, PC, etc...). Saem de cena os sinais de 4-20 mA e outras medidas analógicas : pela rede de dispositivos de campo trafegam dados no formato digital apenas, submetidos ao protocolo que está sendo utilizado. A necessidade de existência do elemento centralizador fica por conta da supervisão e sintonia do sistema, e como já dito, não mais com a finalidade de controle.

Os nós do sistema (dispositivos inteligentes conectados ao Fieldbus), tipicamente executam tarefas simples de sensoriamento, monitoração e atuação, mas quando trabalhando juntos podem perfazer tarefas bastante complexas. Essa evolução nos sistemas de automação é similar à evolução da indústria de informática: a 15 anos tinha-se terminais "burros" conectados a um computador central, hoje o processamento é distribuído entre diversos computadores pessoais e os dados são trocados através de uma rede.



1.7 Sistemas Abertos

Com todas as facilidades da comunicação em rede, protocolos padronizados e dispositivos inteligentes, um outro objetivo parece ser atingível: Sistemas Abertos . Interoperável e intercambiável, um sistema aberto é aquele em que seus componentes podem ser adquiridos de um ou outro fornecedor de acordo com um critério definido, seja preço, qualidade ou qualquer outro e num dado momento, por manutenção, upgrade ou outro motivo, esse componente pode ser trocado por aquele fabricado pelo concorrente sem que quaisquer outras alterações sejam feitas no sistema, ou seja, partes que exercem funcionalidades específicas são compatíveis no que se diz respeito à forma como enviam e recebem informações da rede. Para que isso aconteça, o protocolo precisa ser devidamente documentado e aberto, assim como devem existir mecanismos para garantir a compatibilidade de um determinado dispositivo antes que este seja disponibilizado no mercado.

1.8 PC Control

Paralelo à discussão sobre o uso dos Fieldbus com PLC, caminha a da utilização de PC's em sistemas de automação. É fato que os PLC's têm confiabilidade comprovada, alto MTBF, se adaptam muito bem a ambientes e situações críticas, possuem grande parque de instalações de sucesso, etc... São, por outro lado, sistemas fechados, onde peças de expansão ou reposição precisam ser adquiridos sempre do mesmo fabricante e a integração com outros equipamentos, barramentos de campo ou PLC's, quando não é difícil é impossível.

É fato também que os PC's, os sistemas operacionais e softwares desenvolvidos para esses equipamentos têm crescido constantemente em qualidade e melhorado em custos, o hardware é padronizado e conseqüentemente, o problema de manutenção já é resolvido, além de ter melhorias constantes com a manutenção da compatibilidade, o que facilita as

atualizações (upgrades). Talvez, para se atingir parte da robustez de um PLC, se faça necessário um PC com características industriais como placas com maiores camadas de ouro nos contatos, gabinetes reforçados e com sistema para anular as vibrações, entre outras; o sistema operacional utilizado pode precisar de características de tempo real ou o controle pode precisar ser feito sob um software do tipo SoftPLC. Fatores como esses podem elevar os custos dos PC's a patamares próximos aos dos PLC's.

O que se vê no mercado é que, na média as opiniões convergem para o chavão de que "Tudo depende da aplicação" mas é possível verificar tendências para uma e para outra solução. Aqueles que defendem os PLC's dizem que em soluções onde se precisa de segurança, continuidade no controle e respostas em tempo real não é possível aplicar a solução do tipo PC Control, enquanto que os que defendem a bandeira dos PC's no controle de sistemas de automação são capazes de afirmar que em qualquer aplicação o PC substitui o PLC.

Ponderando sobre as opiniões e observando o que de fato existe nos sistemas implantados, percebe-se que cada solução tem seu lugar, mas naquelas que utilizam barramentos de campo e dispositivos inteligentes, os PC's tornam-se opções muito interessantes, pois, todos os aspectos de confiabilidade e robustez são passados para a esfera do instrumento e da rede e o PC passa a ser mais um elemento de supervisão do que de controle (tarefa que os PC's já fazem confiavelmente em sistemas de automação de todos os tipos), podendo até, em alguns casos, ser desligado sem comprometer o funcionamento do sistema.

1.9 O PC nas áreas de automação

Talvez pela real base instalada, talvez por interesse comercial dos fabricantes ou talvez pela simples dificuldade de mudança de paradigma, algumas áreas da automação se comportam de maneira diferente quanto à aceitação de sistemas com uma ou outra tecnologia. De maneira geral, na automação industrial e de processos existe grande demanda pela tradicional solução com PLC's, a utilização de dispositivos inteligentes é discreta e a de PC's de alguma forma vista com algum preconceito enquanto que na automação predial e residencial os fieldbus são usados em larga escala e, conseqüentemente, os PC's também se aplicam com sucesso.

Sistemas de automação baseados em PC estão atualmente sendo implementados como:

PC industrial com placa co-processadora para as tarefas de PLC / CNC, computador para controle de processo ou sistema de controle combinado com PC.

Os PC's não estão executando tarefas de controle diretamente. Estas, são, normalmente realizadas por cartões co-processadores. A solução puramente baseada em software é uma nova proposta na qual o processador realiza toda a tarefa de automação dentro de um sistema com um único processador.

A Proposta desta solução é a seguinte: menos componentes significa menos problemas.

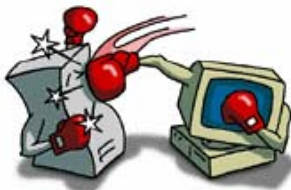
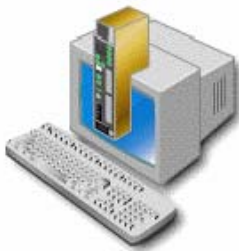
No desenvolvimento de placas de PC's industriais (IPC), placas de interface, fontes de alimentação de baixa tensão, controladores de teclado, controladores de UPS e outros módulos eletrônicos é necessário uma extrema confiabilidade para trabalhar em ambiente industrial agressivo.

Tem havido uma reorientação no que diz respeito ao conceito de PC industrial. O painel de operação está sendo separado do computador. O computador está sendo montado

dentro do painel elétrico. Esta é uma nova proposta que permite muita liberdade de movimento e que tem sido implementada consistentemente com os painéis de controle. O painel de operação (controle e display), constituem uma entidade independente que é mantida separada do controlador. O PC está protegido dentro do painel elétrico e é facilmente acessado. Um encapsulamento para o PC pode ser confeccionado para instalação de seus componentes. O painel de controle tem um perfil elegante com uma grande área para a interface com o usuário, porém com uma profundidade mínima.

1.10 COMPUTADOR INDUSTRIAL VERSUS CONTROLADOR PROGRAMÁVEL

A arquitetura de um controlador programável é basicamente a mesma que um computador de propósito geral. Entretanto existem algumas características importantes que diferem o CP dos computadores. Podemos dizer que todos os CP's são computadores por definição, mas nem todos os computadores são CP's. A diferença está nos métodos de programação, operação, considerações ambientais e manutenção. A figura abaixo ilustra uma comparação entre computadores industriais e CP onde podem ser vistos os pontos fortes e os pontos fracos dos computadores industriais.



Pontos fortes

- Interface Gráfica
- Tempo de Programação
- Não utilizar Hardware Proprietário
- Arquitetura Aberta
- Rede de comunicação TCP/IP
- Simulação do Programa
- Várias Linguagens de Programação
- Comunicação com Supervisorio
- Utilização de vários Hardwares de E/O
- Facilidade de efetuar calculos complexos

Pontos fracos

- Confiabilidade do Sistema Operacional
- Confiabilidade do Microcomputador
- Velocidade de Atualização de E/O (Rack)
- Eventuais Bugs de Software

2. SISTEMAS DE SUPERVISÃO E CONTROLE

2.1 Interface Homem-Máquina

Embora existam diversas tecnologias de controle, deve existir o interfaceamento entre o controle com o comando, proporcionando uma melhor integração do homem com a máquina. Para esta finalidade, utilizou-se com frequência anunciadores de alarmes, sinaleiros chaves seletoras botoeiras, etc..., que nos permitiam comandar ou visualizar estados definidos com ligado e desligado, alto ou baixo, temperatura elevada ou normal, mas não nos permitia visualizar os valores de alto, quando alto, ou normal quando normal. Surgiram então os “displays” e chaves digitais (“thumbwheel switches”). Os “displays” nos permitiam visualizar os valores das variáveis do processo, bem como mudar parâmetros pré definidos, como por exemplo, temporizações através das chaves digitais.

No entanto, este tipo de interface trazia dois problemas claros, o primeiro a dimensão da superfície do painel, que por muitas vezes necessitava de ser ampliada, somente para alojar tantos botões, ou informações que eram necessárias. Com o desenvolvimento das interfaces homem-máquina – IHM - com visores alfanuméricos, teclados de funções e comunicação via serial com o dispositivo de controle, o qual muitas vezes era um computador pessoal – PC - estas traziam consigo os seguintes benefícios:

- Economia de fiação e acessórios, pois a comunicação com o CP seria serial com um ou dois pares de fio trançados, economizando vários pontos de entrada ou saída do CP, e a fiação deste com os sinaleiros e botões.
- Redução da mão-de-obra para montagem, pois ao invés de vários dispositivos, agora seria montado apenas a IHM.
- Diminuição das dimensões físicas do painel
- Aumento da capacidade de comando e controle, pois a IHM pode ajudar em algumas funções o CP, com por exemplo massa de memória para armazenar dados, etc.
- Maior flexibilidade frente a alterações no campo.
- Operação amigável
- Fácil programação e manutenção.

A evolução seguinte foi a utilização de interfaces gráficas ao invés de alfanuméricas.

Quando utilizadas, as interfaces gráficas, em alguns casos mais simples substituem os sistemas supervisórios, ou quando usadas em sistemas de controle, integradas a sistemas supervisórios, estas além das funções das IHM's alfanuméricas já citadas, executam também funções de visualização que aliviam o sistema supervisório para que a performance das funções de supervisão, alarme, tendências, controle estatístico de processo entre outras.

Logo, os softwares que tem a finalidade de servir como uma Interface Homem Máquina, não tem a finalidade de controlar nenhuma parte da máquina ou processo, ou seja, se ocorrer qualquer problema durante a sua execução, não prejudicará a automação da máquina ou processo. Normalmente estes softwares apresentam facilidades de configuração, mas estão limitados em segurança de dados, comunicação em rede, comunicação remota, controles de processo, etc.

Os softwares que possuem as mesmas funções dos softwares IHM, além de poderem efetuar controle, distribuir informações entres estações via rede com performance e segurança, etc., são os softwares do tipo “sistema de supervisão, controle e aquisição de dados”- SCADA. Estes softwares normalmente são mais robustos e confiáveis para aplicações de grande porte e para aplicações distribuídas em varias estações.

Um exemplo de um IHM básico esta mostrado abaixo:



Interface Homem-máquina (IHM)

2.2 Software de Supervisão e Controle

O software de supervisão, localizado no nível de controle do processo das redes de comunicação, é o responsável pela aquisição de dados diretamente dos controladores lógico programáveis - CLP para o computador, pela sua organização, utilização e gerenciamento dos dados. Poderá ser configurado para taxas de varredura diferentes entre CLP's e inclusive, entre pontos de um mesmo CLP.

Os dados adquiridos devem ser condicionados e convertidos em unidades de engenharia adequadas, em formato simples ou de ponto flutuante, armazenando-os em um banco de dados operacional. A configuração individual de cada ponto supervisionado ou controlado, permite ao usuário definir limites para alarmes, condições e textos para cada estado diferente de um ponto, valores para conversão em unidade de engenharia, etc..

O software deve permitir que estratégias de controle possam ser desenvolvidas utilizando-se de funções avançadas, através de módulos dedicados para implementação de funções matemáticas e booleanas, por exemplo. Através destes módulos, poderá ser feito no software aplicativo de supervisão, o controle das funções do processo.

Os dados adquiridos podem ser manipulados de modo a gerar valores para parâmetros de controle como "set-point's". Os dados são armazenados em arquivos de dados padronizados. Estes arquivos poderão ser acessados por programas de usuários para realização de cálculos, alteração de parâmetros e dos seus próprios valores.

O software supervisorio é visto como o conjunto de programas gerado e configurado no software básico de supervisão, implementando as estratégias de controle e supervisão, as telas gráficas de interfaceamento homem-máquina, a aquisição e tratamento de dados do processo, a gerência de relatórios e alarmes. Este software deve ter entrada de dados manual, através de teclado. Os dados serão requisitados através de telas com campos pré-formatados que o operador deverá preencher. Estes dados deverão ser auto-explicativos e possuírem limites para as faixas válidas. A entrada dos dados deve ser realizada por telas individuais, seqüencialmente, com seleção automática da próxima entrada. Após todos os dados de um grupo ser inserido, esses poderão ser alterados ou adicionados pelo operador, que será o responsável pela validação das alterações.

A estratégia de supervisão e controle é desenvolvida com o software básico de supervisão que cria um banco de dados operacional com todos os dados de configuração do sistema. Os dados podem ser referentes a configuração da própria estratégia ou referentes aos pontos supervisionados (ou controlados). Em ambos os casos, o método e recursos utilizados para entrada de dados deve ser composta por ferramentas do tipo "Windows", com menus dirigidos, preenchimento de campos pré-formatados e múltiplas janelas.

Os dados da estratégia são gerais, afetando todo o banco, como por exemplo, a configuração de impressoras, os tipos de equipamentos conectados, as senhas, etc.. Os dados referentes aos pontos são individuais e abrangem os "TAG" (variáveis de entrada/saída - I/O - ou internas), as descrições, os limites de alarme, a taxa de varredura, etc.. Alterações podem ser realizadas com o sistema "on-line" (ligado ou à quente). Após a estratégia configurada, o software básico deve executar, gerenciar e armazenar o resultado de cálculos e operações realizadas, o estado dos pontos e todas as informações necessárias neste banco de dados.

O conjunto de telas do software de supervisão deve permitir os operadores, controlar e supervisionar completamente toda a planta. As telas deverão ser organizadas em estrutura hierárquica do tipo árvore, permitindo um acesso seqüencial e rápido. A seguir, é descrito as principais telas que o aplicativo deve conter:

- Telas de visão geral: são telas que apresentarão ao operador uma visão global de um processo, sob visualização imediata na operação da planta. Nestas telas são apresentados os dados mais significantes à operação e objetos que representam o processo. Os objetivos devem ser dotados de características dinâmicas, representando o estado de grupos de equipamentos e áreas do processos apresentado. Os dados devem procurar resumir de forma significativa os principais parâmetros a serem controlados (ou monitorados) do processo específico;
- Telas de grupo: são telas representativas de cada processo ou unidade, apresentando objetos e dados de uma determinada área de modo a relacionar funções estanques dos processos. Os objetos devem ser dotados de características dinâmicas representando o estado e/ou condição dos equipamentos da área apresentada. Os dados apresentados devem representar valores quantitativos dos parâmetros supervisionados (ou controlados). As telas de grupo também possibilita ao operador, acionar os equipamentos da área através de comandos do tipo abrir/fechar ou ligar/desligar. Além disso, o operador poderá alterar os parâmetros de controle ou supervisão, tais como "set-point's", limites de alarme, modos de controle, etc.;
- Telas de detalhe: são telas que atendem a pontos e equipamentos controlados (ou monitorados) individualmente. Serão compostas, quando possível, por objetos com características dinâmicas, representando o estado do equipamento. Os dados apresentam todos os parâmetros do ponto supervisionado (ou monitorado). As telas devem possibilitar ao operador alterar os parâmetros do equipamento, seus limites, os seus dados de configuração, etc.;
- Telas de malhas: são telas que apresentam o estado das malhas de controle. Todas as telas devem apresentar os dados das variáveis controladas exibidas, como "set-point's", limites e condição dos alarmes, valor atual e valor calculado, etc., em forma de gráfico de barras e em valores numéricos;
- Telas de tendência - histórica e real: são telas normalmente padrão do software básico de supervisão. Estas telas apresentam várias (em média seis) variáveis simultaneamente, na forma gráfica, com valores coletados em tempo real ("on-line"), na forma de tendência real e na forma histórica "off-line" - valores de arquivos pré-armazenados em disco. Estas tendências podem ser apresentadas em forma de gráficos ou em forma tabular, com os últimos valores coletados para cada variável;
- Telas de manutenção: são compostas por informações de problemas, alarmes, defeitos e dados de manutenção das diversas áreas referentes ao processo e equipamentos destes, incluindo o próprio sistema de controle. As informações são do tipo

histórico de falhas, programa de manutenção dos equipamentos (corretiva e preventiva), e informações gerais dos equipamentos (comerciais, assistências técnica, etc.). O histórico de falhas por equipamento ou área fica armazenado em arquivos no banco de dados do software de supervisão, possibilitando o tratamento destas informações através de telas orientativas à manutenção, ou através de programas de usuário para estatísticas de utilização e defeitos.

O software básico de supervisão possui um módulo para desenvolvimento de relatórios. Criados em formatos padrão, para os relatórios do tipo históricos, permitem ao operador a escolha de quais variáveis deseja visualizar. Os dados podem ser apresentados nas telas das estações com campos de identificação para "TAG", data, hora e descrição do ponto. Os relatórios poderão ser solicitados manualmente pelo operador e destinados para impressoras ou terminais de vídeo. Os dados históricos são armazenados em arquivos de modo que podem ser acessados pelos programas de relatórios, para serem trabalhados e apresentados à operação. Deste modo, os arquivos podem ser armazenados em meios magnéticos para utilização futura.

Dentro deste perfil de aplicativos de supervisão, encontram-se vários fabricantes como: WIZCON e WIZFACTORY- PC Soft International, Inc., RSI - Rockwell Automation, LabVIEW - National Instruments e ELIPSE SCADA - Elipse Software Ltda.

O Elipse SCADA é um software para criação de aplicativos de supervisão e controle de processos nas mais diversas áreas, tais como: química, automação predial, manufatura, elétrica, segurança, laboratórios de testes, saneamento e máquinas. Desenvolvido em um ambiente avançado, orientado a objetos e de fácil configuração, permite a visualização e acionamento de variáveis, bancos de dados, relatórios, receitas e conectividade com todo tipo de sistemas, inclusive via Internet. Este aplicativo oferece sofisticados recursos que otimizam o tempo de desenvolvimento e a manutenção dos sistemas:

- Interface clara, lógica e intuitiva;
- Conectividade com a maioria dos equipamentos disponíveis no mercado ou mesmo com outros aplicativos Windows;
- Biblioteca gráfica para criação de telas; Suporte à rede e arquitetura cliente/servidor;
- Configuração e reconhecimento de alarmes;
- Relatórios formatados, graficamente customizados pelo usuário;
- Registro de dados em disco e análise histórica;
- Receitas que permitem a programação de valores para o envio ao processo;
- Scripts que permitem a criação de rotinas exclusivas, definindo lógicas e criando seqüências de atitudes através de uma linguagem de programação interativa, personalizando ao máximo o aplicativo;
- Suporte a banco de dados via ODBC (Open Data Base Connectivity) - Access, SQL Server, Oracle, dBase, etc.;
- CEP (Controle Estatístico de Processos);
- Módulo matemático para a formulação de equações;
- Controle de acesso por nível de usuário;
- Acesso remoto via Internet;
- Captura, registro e transmissão digital de imagens;
- Aquisição de eventos com precisão de 1ms;
- Criação de instrumentos virtuais.



O aplicativo Elipse SCADA contém vários aplicativos que possibilitam a adequação do software, ao tamanho e complexibilidade do processo. A versão recomendada para o caso específico desse projeto é o Elipse MMI (MAN MACHINE INTERFACE). Esta versão é indicada para aplicações de médio porte, onde é necessária a coleta de dados e o tratamento de informações. Este é um software de supervisão completo com banco de dados proprietário, relatórios formatados, históricos, receitas, alarmes e controle estatístico de processos. Este aplicativo não possui ODBC, não permite Cliente NetDDE e não é Cliente de Rede, isto é, não possui aplicações remotas.