

DESENVOLVIMENTO DE SONAR ATMOSFERICO ATIVO PARA DETERMINAÇÃO DE POSIÇÃO

Antonio Augusto Russo

Helio Mitio Morishita

Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia Naval e Oceânica, 05508-900, São Paulo, SP, Brasil.

Resumo:

A medida da posição de um objeto móvel é de fundamental importância para o desenvolvimento de sistemas de controle. Entretanto, ainda não estão facilmente disponíveis métodos para efetuar essas medidas em escalas de modelos. Este projeto diferencia-se no meio de propagação utilizado – o ar - e não a água. Essa escolha permite uma maior flexibilidade na utilização do sistema possibilitando o seu uso em ambiente terrestre, além é claro, da simplificação da montagem pois os componentes acústicos (alto-falantes e microfones) são, via de regra, menos custosos que seus equivalentes submarinos. Assim, pode-se encarar este projeto como uma adaptação dos sistemas de posicionamento já consagrados para uma aplicação específica, de baixo custo que, todavia, possa ser utilizada como instrumento de confiança.

Palavras-chave: Ultra-som, Sonar, Localização.

1. Introdução:

A determinação das características hidrodinâmicas de veículos oceânicos é efetuado, normalmente, realizando ensaios com modelos reduzidos. E muitas vezes durante os ensaios, é necessário medir a posição e a velocidade do modelo para determinar, adequadamente, as características dinâmicas da embarcação. A medida de movimentos em tanques de laboratório hidrodinâmicos pode ser efetuada utilizando instrumentos específicos. No entanto, estes mesmos instrumentos podem ser inadequados ou mesmo inviáveis quando se deseja efetuar ensaios com modelos maiores em águas restritas como em lagos ou represas. A medida de posição de modelos reduzidos de veículos oceânicos não é simples, mormente quando há necessidade de compatibilizar a natureza do ensaio, precisão e custo. Mas, atualmente, os sistemas digitais de alta velocidade permitem desenvolver medidores de posição, com precisão requerida, aplicando os princípios clássicos como propagação acústica. O Departamento de Engenharia Naval e Oceânica da Escola Politécnica da Universidade de São Paulo, em particular, tem interesse no desenvolvimento deste tipo de sensor, pois ele é de extrema utilidade nos ensaios com modelos no tanque do seu Laboratório.

2. Princípios de Funcionamento:

Todos os métodos de medida de distância que não envolvem vínculos mecânicos baseiam-se na medida do tempo de propagação de certo tipo de energia entre dois pontos. Radares e GPS utilizam ondas eletromagnéticas; Sonares empregam ondas acústicas. Tais

equipamentos também diferem na forma como as medidas de distância são transformadas em medida de posição absoluta.

Então, para medir uma distância, deve-se conhecer a velocidade de propagação da forma de energia escolhida, no meio em que está se propagando e então medir o tempo decorrido entre a emissão da onda (eletromagnética ou acústica) em um dos pontos e sua recepção no outro ponto. Sabendo que as distâncias típicas envolvidas são da ordem de dezenas de metros apresenta-se imediatamente a escolha de sinais acústicos de alta frequência para operação do equipamento. Para medir a distância, foi visto que era necessário apenas medir um intervalo de tempo e conhecer a velocidade de propagação. O coração do projeto, portanto, consiste em desenvolver um dispositivo que possa medir intervalos de tempo com precisão e resolução adequadas. Foi escolhido uma placa de interface para PC como solução. De posse de todas as limitações e características acima consideradas bem como das especificações desejadas chegou-se aos seguintes valores:

- Freq. de amostragem Maior que 10 Hertz (>10Hz)
- Resolução Menor que 10 centímetros (<10cm)
- Precisão Menor que 5% do valor medido (< 5%)
- Alcance Aproximadamente cem metros (~100m)

Para efetuar a medida dos intervalos de tempo foi escolhido o uso de contadores digitais (*timers*) são controlados por uma interface comum, através do barramento de um PC. Foram escolhidos *timers* de 16 bits devido a sua flexibilidade e facilidade de implementação. A placa de interface contém oito *timers* independentes e um *prescaler*. Tendo disponíveis oito sensores pode-se agrupá-los em dois conjuntos de quatro de tal forma que cada grupo possa efetuar a medida com redundância, e portanto com maior confiabilidade. A existência de dois conjuntos de sensores possibilita ainda a medida de dois pontos simultaneamente; isso é particularmente interessante em engenharia oceânica pois permite medir o aproamento de um modelo instalando sensores nas extremidades do modelo. A figura 1 mostra o ciclo de aquisição de cada timer, mostrando que além do tempo decorrido na propagação existem outros atrasos que devem ser considerados.

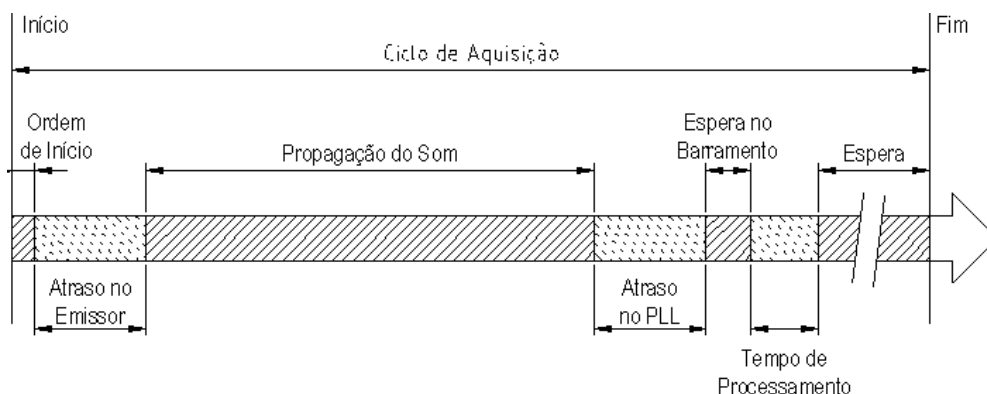


Figura 1 – Tempos e eventos de um ciclo de aquisição

O sinal acústico consiste basicamente de um tom de 25kHz, de potência suficiente para que atinja com intensidade adequada. O receptor deve ser capaz de discernir este sinal do ruído ambiente; para tal deve conter um filtro PLL, sintonizado na frequência de operação. Como todo circuito real, tanto o emissor como os receptores introduzem atrasos que devem ser compensados posteriormente. Em particular, no receptor, este atraso deve-se principalmente ao atraso do PLL.

O problema que se impõe agora é determinar a posição a partir de medidas convenientes de distâncias. Entende-se por determinar a posição de um ponto como encontrar as coordenadas que definem este ponto em um certo sistema de referência fixado previamente. É possível mostrar que é necessário medir $N+1$ distâncias para poder localizar, univocamente, um ponto em um sistema N -dimensional. A distribuição dos sensores no espaço de trabalho é, *a priori*, completamente livre. Entretanto, para minimizar os efeitos de mal-condicionamento numérico sugere-se distribuir os sensores sobre os vértices de um polígono regular (TETLEY et al., 1986), com um deles definindo a origem do sistema de coordenadas. Particularmente, neste projeto, utiliza-se quatro sensores, dispostos sobre um quadrado, com um sensor definindo a origem e outros dois sensores definindo os eixos coordenados conforme mostrado na figura 2.

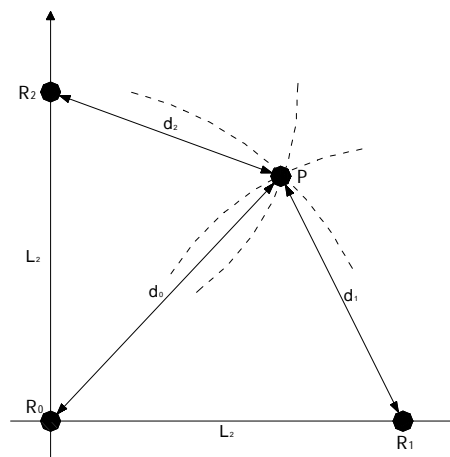


Figura 2 – Determinação da posição através das medidas de distâncias

3. Algoritmos:

3.1. Algoritmo de Localização

Para poder determinar a posição do móvel é necessário, inicialmente, estabelecer a distribuição espacial dos sensores (constelação) $\{\mathbf{P}_i\}$. Fixada a constelação e conhecidas as distâncias entre cada sensor e o móvel, $\{\mathbf{R}_i\}$, pode-se então calcular a posição. Aqui será mostrado o método para o cálculo da posição em duas dimensões; entretanto pode-se mostrar que o algoritmo é válido no caso geral em N dimensões, com M sensores ($M > N$), e em particular, para o caso tridimensional (localização espacial).

Especificamente, neste trabalho, a configuração utilizada é dada por uma constelação de quatro sensores, situados no mesmo plano. Pode-se admitir, sem perda de generalidade, que os sensores estejam localizados no plano de solo ($Z=0$). O móvel, por hipótese, também pertence a este plano. Tais suposições podem parecer restritivas à primeira vista, entretanto não se perde de vista o objetivo inicial do projeto que é um sistema de localização plano.

Formalização do Problema:

Sejam, então as posições dos sensores, $\{\mathbf{P}_i\}$, a posição do móvel \mathbf{X} ; em um certo sistema de coordenadas e o conjunto das distâncias que separam os pontos \mathbf{P}_i do ponto \mathbf{X} :

$$\mathbf{P}_i = [X_i \quad Y_i \quad Z_i] \quad e \quad \mathbf{X} = [X \quad Y \quad Z]$$

Para resolver cada sistema emprega-se o Método de Newton-Raphson generalizado para o caso multivariável:

$$\mathbf{X}_i^{(k+1)} = \mathbf{X}_i^{(k)} - J_i(\mathbf{X}^{(k)})^{-1} \cdot f_i(\mathbf{X}^{(k)})$$

Onde:

$$\begin{aligned} \mathbf{X}_i & \quad \lim_{k \rightarrow \infty} \mathbf{X}_i^{(k)} \quad \text{é a solução,} \\ \mathbf{X}_i^{(k)} & \quad \text{é a Aproximação da solução do sistema } i \text{ após a } k - \text{ésima iteração,} \\ f_i(\mathbf{X}^{(k)}) & \quad \text{é a Função erro de aproximação } \rightarrow f_i(\mathbf{X}_i) = \mathbf{0}, \\ J_i(\mathbf{X}^{(k)}) & \quad \text{é o Jacobiano da função erro.} \end{aligned}$$

Por exemplo, para o sistema Σ_3 , composto do conjunto de todos os sensores exceto o terceiro. (Sensores 0, 1 e 2, nas posições \mathbf{P}_0 , \mathbf{P}_1 e \mathbf{P}_2 , respectivamente)

$$f_3(X, Y, Z) = \begin{bmatrix} (X - X_0)^2 + (Y - Y_0)^2 + (Z - Z_0)^2 - R_0^2 \\ (X - X_1)^2 + (Y - Y_1)^2 + (Z - Z_1)^2 - R_1^2 \\ (X - X_2)^2 + (Y - Y_2)^2 + (Z - Z_2)^2 - R_2^2 \end{bmatrix} \quad f_3(\mathbf{X}) = \begin{bmatrix} (\mathbf{X} - \mathbf{P}_0) * (\mathbf{X} - \mathbf{P}_0) - R_0^2 \\ (\mathbf{X} - \mathbf{P}_1) * (\mathbf{X} - \mathbf{P}_1) - R_1^2 \\ (\mathbf{X} - \mathbf{P}_2) * (\mathbf{X} - \mathbf{P}_2) - R_2^2 \end{bmatrix}$$

A convergência do método é assegurada pela convexidade do espaço das soluções. Quanto à unicidade da solução tem-se que, como o sistema é quadrático, podem existir duas soluções, simétricas em relação ao plano definido por $\{\mathbf{P}_i\}$; entretanto, no caso plano (\mathbf{X} pertence àquele plano), as duas soluções coincidem. Resolvendo, então, a todos os sistemas e chamando de \mathbf{X}_i a solução de Σ_i , pode-se aplicar métodos estatísticos a fim de estimar os desvios associados às medidas. A estatística utilizada calcula médias e desvios-padrão da medida de posição e infere o desvio máximo através de testes de hipóteses.

3.2. Algoritmo de Calibração

Como pode ser visto na Fig.1 existe uma diferença entre o tempo total medido e o intervalo da propagação propriamente dita. Para estimar essa diferença assim como estimar a velocidade de propagação o equipamento deve ser calibrado antes do uso. Esta diferença corresponde ao tempo de trânsito do sinal nos circuitos do receptor, do emissor, tempo de reconhecimento do sinal, etc. e pode ser considerado fixo. Com isso podemos equacionar:

$$t_{medido} = t_{fixo} + t_{propagação} \quad \text{onde: } t_{propagação} = \frac{d}{v_{som}}$$

Desta forma, pode-se estimar os tempos fixos assim como determinar a velocidade de propagação do som no ambiente, minimizando os efeitos da pressão atmosférica, temperatura e umidade sobre o valor velocidade de propagação.

O procedimento de calibração consiste em medir os tempos totais para uma série de distâncias conhecidas e, desta forma tendo então um relação linear simples entre tempo medido e distância pode-se facilmente, através de algumas medidas, estabelecer uma regressão linear da forma: $y=ax+b$. Com isso pode-se estimar os desvios associados às medidas dos tempos, levando em conta as incertezas nos parâmetros como velocidade do som, posição dos sensores, tempos fixos, etc. Deste modo, pode-se determinar a classe de precisão do equipamento, que como em todo instrumento de medida, deve ser conhecida a fim de se estimar a confiabilidade da medida de forma que esta tenha realmente significado no ensaio efetuado.

4. Hardware:

O emissor de ultra-som consiste em um oscilador controlado, PLL, que gera um sinal de frequência conhecida (25kHz) e um amplificador de potência capaz de excitar convenientemente o transdutor. O oscilador é controlado por um sinal digital externo, enviado pela interface de contagem, que habilita a oscilação. Quanto ao amplificador de potência, este deve ser capaz de operar a uma frequência razoavelmente alta (25kHz) e de transferir potência suficiente para que a intensidade do som junto aos receptores seja adequada. Também deve operar como um filtro passa-baixas, evitando desperdiçar potência em faixas do espectro que não são de interesse à operação do sistema. O sinal gerado pelo oscilador é uma onda quadrada, com 50% de ciclo ativo, e juntamente com a característica dinâmica do transdutor escolhido causaria o aparecimento de harmônicas indesejáveis no sinal emitido. Para tanto o amplificador é sintonizado de modo a atenuar frequências acima de 25kHz. O transdutor escolhido é um *tweeter* piezelétrico de 50Wrms e frequência de corte superior em 12kHz. O *tweeter* opera além da frequência de corte, mas ainda consegue entregar ao meio uma potência razoável para os propósitos do sistema.

Similarmente ao circuito do emissor, o receptor é constituído de transdutor, amplificador e de um PLL. Neste caso o PLL atua como filtro sintonizado. O transdutor utilizado é um microfone de eletreto comum, que mesmo operando além de sua frequência de corte, apresenta resposta suficiente às necessidades do sistema. A atenuação devido ao corte é compensada pelo amplificador que opera como filtro passa-altas sintonizado na mesma frequência de operação do emissor (25kHz).

O módulo de contagem consiste de uma placa padrão ISA, de 8 bits, que realiza a contagem dos tempos entre os disparos dos PLL e controla o circuito do emissor de ultra-som. Analisando o funcionamento do módulo de contagem com maior detalhamento, vê-se que este é composto por três blocos: Interface com o barramento, Módulos de contagem, e Controle de Disparo (AUSTERLITZ, 1991). A escolha do padrão ISA, já considerado obsoleto na indústria, em detrimento de uma interface PCI deve-se mormente a facilidade de projeto e implementação, assim como leva em conta a disponibilidade de PC's mais antigos, que não contam com aquele tipo de interface já que a demanda por processamento não é crítica (RUSSO, 1999). A figura 3 detalha o interrelacionamento entre os módulos.

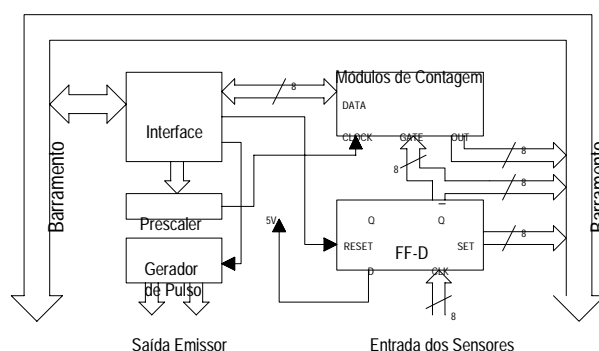


Figura 3 – Interrelação entre os blocos componentes da interface

Basicamente a placa contém oito *timers* independentes, de 16 bits, que são utilizados para medir os tempos decorridos entre a emissão do pulso e sua recepção. O clock que excita tais *timers* provém de um *prescaler*, completamente programável por *software*. A lógica de controle do travamento dos contadores é composta por um vetor de oito *flip-flops* tipo D, sensíveis a borda. A função destes *flip-flops* é reconhecer a borda do sinal vindo do PLL e travar o contador correspondente. Como visto, cada contador está habilitado quando a sua

entrada GATE está em 1. Todos os *flip-flops* são idênticos em funcionamento e operam como um bloco uno em relação as operações de I/O com o barramento.

4.1. Funcionamento:

Antes da operação propriamente dita é necessário inicializar o sistema, escrevendo as palavras de controle (*Control Words*). São nove *timers* independentes onde um será usado como divisor de frequências e os outros oito para a medida de intervalo de tempo. O *prescaler* também deve ser inicializado Para isso utiliza-se um dos *timers* operando de modo a dividir a frequência do *clock* do barramento por um fator conveniente. Essa escolha é determinada pelo compromisso entre alcance máximo e resolução da medida.

$$\begin{array}{ll} \text{Resolução :} & \text{Alcance Máximo :} \\ \Delta d_{\min} = \left(\frac{f_{\text{clock}}}{N_{\text{prescaler}}} \right)^{-1} \cdot v_{\text{som}} & R_{\max} = \Delta d_{\min} \cdot 2^{16} \end{array}$$

O ciclo de contagem é iniciado resetando todos os *timers*. Em seguida uma operação de escrita faz com os *flip-flops* da lógica sejam resetados e os respectivos *timers* habilitados. Iniciada a contagem, os *timers* começam a contar, para baixo, com uma frequência determinada pelo *prescaler*. Em seguida são acionados os geradores de pulso para o emissor. Então, quando o sensor recebe o pulso sonoro, o PLL interno gera um sinal (uma borda de subida), que faz com que o *flip-flop* correspondente mude de estado e conseqüentemente trave o *timer*. Quando o sistema operacional determina o próximo ciclo os valores registrados nos *timers* são lidos, juntamente com indicadores de estado de operação (*status flags*) e o fator de divisão do *prescaler*. Os *status flags* reportam eventuais erros de operação do sistema, tais como perda de um dos sensores ou localização além da região de máximo alcance do equipamento.

5. Software:

Optou-se por desenvolver o programa no ambiente Microsoft Visual Studio® 97. Este ambiente propicia alto grau de integração entre o compilador e o sistema operacional, além de disponibilizar inúmeras bibliotecas e ferramentas que simplificam o desenvolvimento da Interface Gráfica ao Usuário (GUI). Optou-se por não utilizar o suporte MFC (*Microsoft Foundation Classes*) oferecido a fim de maximizar a performance do programa; utilizado chamadas diretas para a API Win32. Isso permite que todos os recursos de interface do ambiente Win32 (Win95™ ou WinNT™) estejam disponíveis, sem entretanto acrescentar o *overhead* típico das soluções MFC.

5.1. Classes de Dados e Rotinas Numéricas

A razão do emprego quase abusivo da notação vetorial na descrição do problema fica evidente quando é mostrado a transposição das equações anteriores em código C++. Foram definidas classes de objetos tipo **Point**, representando Pontos ou Vetores no espaço 3D e tipo **Matrix**, representando Matrizes reais, de dimensão 3x3. Essas classes foram escritas de forma a permitir operações internas, escalares e relativas entre si tornando quase imediata a implementação do algoritmo em linguagem de programação. Com tais classes de dados, os problemas de Álgebra Linear, no espaço cartesiano \mathbb{R}^3 , podem ser representados de forma simples e elegante, fazendo com que não perca tempo com o entendimento de detalhes técnicos (RUSSO, 1999); deixando o algoritmo propriamente dito o mais parecido possível com a notação algébrica.

5.2. Interface com Hardware (device driver):

Em uma arquitetura por camadas a aridez de tais comandos deve ser encapsulada em comandos de nível mais alto, que agreguem as funções mais comuns. O objetivo final, é praticamente o mesmo descrito na seção anterior: fazer com que o programa principal seja intuitivo e simples, afastando os detalhes e incorporando-os aos respectivos subsistemas. Para isso devem ser criadas API's para as tarefas mais comuns como ajuste da frequência de contagem, ajuste da frequência do *prescaler* (alcance máximo) e de habilitação dos sensores utilizados. Deve também prover rotinas para inicializar a placa de interface, resetar os contadores e interagir com o sistema operacional.

A malha de repetição fica implícita, sendo sincronizada por *timers* internos do SO. Tal técnica é chamada de orientação por eventos, onde a execução de trechos de código é iniciada de acordo com a circunstancia e não necessariamente de modo seqüencial. Disso resulta um código final mais simples e transparente ao programador.

5.3. Interface Gráfica com o Usuário:

Inicialmente o programa apresenta a interface (*main window*), onde são representados um sistemas de coordenadas e a posição do conjunto de sensores. Nesta janela estão também o menu de controle das funções do sistema e os indicadores de estado (barra de *status*). O estilo das janelas de interface também segue estritamente o padrão recomendado pelo SDK para nomes de funções, teclas de atalhos, posicionamento de objetos gráficos etc. A interface é completamente intuitiva simples e amigável, necessitando pouco ou nenhum conhecimento técnico para operação, como pode ser visto na figura 4.

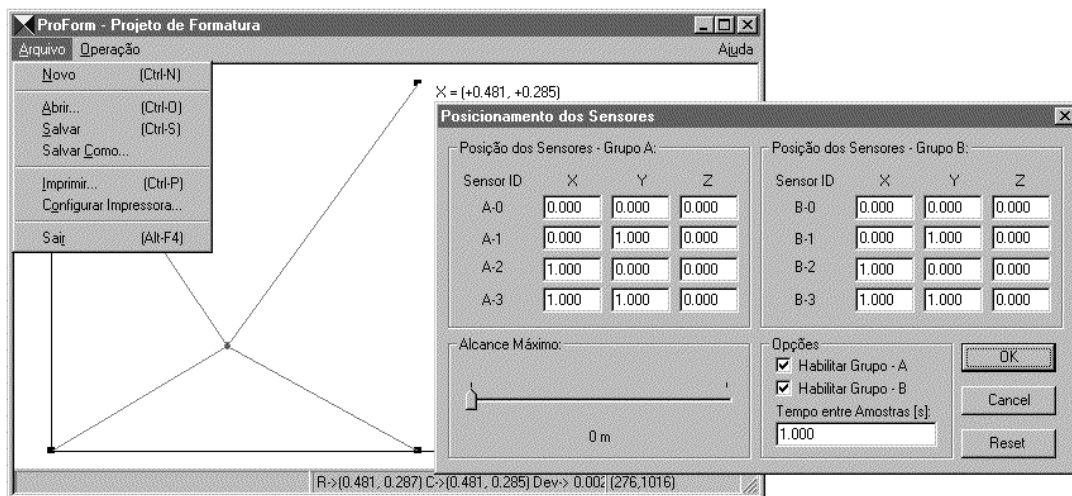


Figura 4 – Interfaces com o Usuário

5.4. Funcionamento

O programa se divide em três camadas (*layers*) diferentes, segundo recomendado pelo SDK Win32: a) Interface ao Usuário; b) Implementação do Algoritmo e c) Interface com *Hardware*. Cada camada é executada em uma sub-tarefa (*thread*) diferente, de modo a otimizar a divisão do tempo de processamento e evitar atrasos na sincronização da leitura da placa. A sub-tarefa c) é, ainda, executada com prioridade superior às tarefas comuns a fim de garantir que nenhuma outra tarefa menos crítica será executada em seu detrimento.

O *software* de controle e interfaceamento do sistema foi totalmente escrito em C++, totalmente orientado a objetos. Quando a aquisição é iniciada o programa instala um *timer* no Sistema Operacional cujo intervalo é o tempo entre amostras e ajusta a frequência do *prescaler* da placa de forma a garantir o alcance máximo na faixa desejada. Em seguida, a

cada intervalo correspondente ao evento do *timer*, é realizada a leitura dos registros da placa correspondentes aos sensores habilitados e os dados lidos são enviados para o algoritmo. Aqui convém tomar cuidado com o sentido da palavra *timer* em cada contexto onde é usada. *Timer* pode ser tanto qualquer um dos módulos de contagem existentes na placa (*hardware*) como também pode ser o dispositivo virtual que o Sistema Operacional oferece que dispara certa tarefa a cada intervalo determinado, em tempo real (*software*).

$$d_i = v_{som} \cdot \left[\left(\frac{f_{barramento}}{N_{prescaler}} \right)^{-1} \cdot N - t_{fixo} \right] \quad \text{onde} \quad \left\{ \begin{array}{l} N = \text{complemento 2} \\ \text{do valor lido no 8254} \end{array} \right.$$

Após a leitura dos registradores (*Hardware Timers*) na placa os valores são convertidos em tempo, de acordo com a frequência do *prescaler*, corrigidos descontando- os tempos fixos e então convertidos em distâncias.

Com todas as distâncias d_i calculadas, o algoritmo pode, então, calcular a posição do objeto. Após o cálculo da posição o objeto é desenhado na área cliente da janela principal, assim como linhas entre os sensores e o ponto, representando as distâncias medidas. As coordenadas do ponto **X** são mostradas no espaço ao lado da representação do ponto e na barra de status, juntamente com a estimativa do desvio padrão da medida (desvio em X e desvio em Y).

6. Conclusão:

Este trabalho destaca-se dos demais sistemas de localização principalmente pelo custo. Foi gasto, até presente momento, cerca de R\$ 1.000,00 na execução do primeiro protótipo. Entretanto metade deste custo é relativo ao projeto do circuito impresso de onde pode-se afirmar que o sistema completo, custe aproximadamente R\$1.000,00 (sem o microcomputador). Outro diferencial é a portabilidade do sistema: O módulo móvel é extremamente leve, possibilitando a utilização em modelo em escala reduzida.

Outra característica chave deste sistema é a flexibilidade do uso. Com o emprego de transdutores adequados e/ou específicos, pode-se adaptar o sistema para operar em uma área mais ampla, ou até mesmo como SONAR submarino. Da mesma forma é possível alocar os receptores no móvel afim de satisfazer alguma possível exigência de construção, etc.

Pretende-se, em futuro próximo, realizar ensaios comparativos a fim de conferir a classe de precisão do equipamento. A comparação será feita com dados adquiridos através de um equipamento RTK-DGPS, com precisão de 5mm. Após esta conferência o sistema passará a fazer parte do instrumental do laboratório do Depto. de Eng. Naval e Oceânica da EPUSP.

Bibliografia:

- AUSTERLITZ, H., 1991; Data Acquisition Techniques Using Personal Computers, Academic Press, San Diego
- MILNE, P. H., 1983; Underwater Acoustic Positioning Systems, Gulf Publishing Company; Houston.
- RUSSO, A. A., 1999; Determinação de Posição Através de Sonar Atmosférico Ativo; Trabalho de Graduação apresentado à Escola Politécnica da USP.
- TETLEY, L. e CALCUTT, D., 1986; Electronic Aids to Navigation, Edward Arnold.
- TOMPKINS, W. J. e WEBSTER, J. G., 1988; Interfacing Sensors to the IBM® PC, Prentice Hall, New Jersey