

# Automação para obtenção de dados de uma experiência de Física: 2ª Lei de Newton

(Data acquisition of a physical experiment: The 2nd Newton Law)

Alejandro R.G. Ramirez<sup>1</sup>, Milton José Cinelli e Adriano Mansur Irigoite

Universidade do Vale do Itajaí, Campus São José, São José, SC, Brasil

Recebido em 29/3/2005; Revisado em 18/8/2005; Aceito em 23/8/2005

Neste trabalho é apresentada uma solução automatizada da experiência sobre a 2ª lei de Newton do módulo da Phywe. No procedimento proposto no manual, a quantidade de dados a ser obtida é relativamente grande e os alunos perdem muito tempo na coleta e ajuste dos dados. Com o novo sistema projetado, o aprendiz focaliza sua atenção nos conceitos físicos envolvidos na experiência fazendo apenas poucos ajustes no controle das variáveis. A solução adotada é baseada em trabalhos que mostram o uso da porta *joystick* como meio de aquisição digital de dados sem necessidade de construir um hardware específico. O controle da experiência, via porta *joystick*, foi desenvolvido usando dois sensores ópticos e um disco perfurado.

**Palavras-chave:** demonstração da lei de Newton, aquisição de dados, porta de *joystick*.

In this work, an automatized solution of the second Newton's law from the Phywe system is presented. Following the instructions found in the Phywe's guidebook, a relatively large amount of data must be obtained, and the students waste much time in collecting and fitting data. With the new design, the students can focus on the physical concepts and make only few adjustments on some variable quantities. The found solution is based in works that have demonstrated the joystick gate are quite useful for digital data acquisition without the need to build up an specific hardware. The proposed solution was developed using two optical detectors and a drilled disk.

**Keywords:** Newton's law demonstration, data acquisition, joystick gate.

## 1. Introdução

No laboratório de Física II há diversas experiências didáticas, e entre elas uma sobre as leis de Newton, a qual é usada pelos alunos de Engenharia para redescobrir e comprovar a 2ª Lei de Newton ( $F = md^2x/dt^2$ ) [1-3]. Nessa experiência medem-se as distâncias percorridas por um carrinho sobre um trilho de ar e os respectivos tempos usando leitores eletrônicos de passagem com incerteza de um milésimo de segundo ( $\pm 0,001$  s). Esses dados são anotados manualmente em tabelas e depois é feita a representação gráfica em papel milimetrado ou com auxílio de software comercial (Excel, Origin etc). Do gráfico linearizado ( $x$  vs.  $t^2$ ) pode-se obter a aceleração do movimento sendo a força externa e massa constantes.

Entretanto, normalmente a quantidade de dados necessários para verificar e redescobrir a 2ª Lei de Newton deve ser relativamente grande o que implica em ajustes de posições dos sensores de passagem de forma repetitiva tornando o experimento pouco atrativo aos

alunos. Além disso, o tempo disponível para as coletas de dados geralmente não é suficiente, o que prejudica o ensino/aprendizado.

Neste artigo, mostra-se a possibilidade de automatizar esse experimento de Física usando-se poucos materiais. Em particular, utilizando dois sensores ópticos de passagem, um disco perfurado e um microcomputador com porta de jogos (*joystick*), tornou-se possível realizar a aquisição de dados do experimento de modo automático. Tornou-se também atrativo ao aluno, pois este novo sistema mostra o gráfico da evolução temporal da distância percorrida pelo carrinho de maneira instantânea. Isto é, aparecem na tela do microcomputador os pontos do gráfico ( $x$  vs.  $t^2$ ) na medida em que o carrinho se desloca ao longo do trilho. Desse gráfico é possível remontar os valores numéricos da velocidade e da aceleração.

Cabe salientar que essa experiência também permite a verificação experimental da variação da aceleração sob ação de força variável, mantendo a massa do carrinho constante.

<sup>1</sup>E-mail: ramirez@univali.br.

## 2. Automação

A automação do experimento implicaria que os sinais provenientes dos sensores deveriam ser digitalizados, usando, por exemplo, um conversor analógico-digital (ADC) e sendo, desta forma, armazenados em um computador pessoal (PC). Contudo, no decorrer deste trabalho, percebeu-se que não havia a necessidade de se construir um circuito específico (*hardware*), pois se poderia ler o valor dos sensores diretamente na forma digital usando alguma entrada já existente no PC, seja pela porta paralela ou pela porta serial.

Decidiu-se diferenciar um pouco e utilizar uma outra porta de entrada de dados no computador, sendo algo não tão novo, mas diferente: a porta de jogos (*joystick*), pelo fato de que atualmente a maioria das placas mães já esta vindo com uma placa de som embarcada, sendo acoplada a porta de jogos. Desta forma, a existência de uma porta de *joystick* em um PC seria facilmente encontrada livre sem que houvesse um mouse (serial) encaixado ou uma impressora (paralela) sendo utilizada. Antecedentes destas aplicações podem ser encontrados em [4-6].

## 3. Porta de jogos

A porta de joystick utiliza um conector do modelo DB15 (fêmea) e normalmente se encontra em uma placa de som, embarcada na placa mãe. Na Fig. 1, são mostrados os pinos deste conector, ver [6].

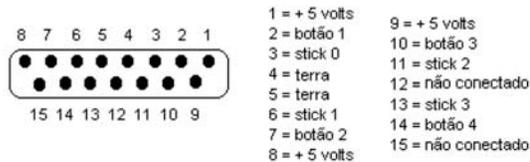


Figura 1 - Diagrama de pinagem no conector do joystick DB15

Pode-se observar na figura 1 que esta porta tem quatro entradas (botões 1, 2, 3 e 4) que são utilizadas para receber informações digitais binárias (0 ou 1) e quatro entradas (*sticks* 1, 2, 3 e 4) usadas para receber dados analógicos. O PC lê e escreve os dados desta porta no endereço 201h ou 209h, dependendo da configuração do computador. Os quatro primeiros bits indicam o estado lógico dos botões e os quatro últimos bits mostram o estado dos osciladores usados para medir o valor da corrente existente nas entradas analógicas da porta de jogos, o qual requer um processamento adicional.

No entanto, os estados lógicos das entradas digitais são interpretados diretamente pelo processador. Essas entradas estão conectadas por meio de um resistor ao potencial de 5 V, apresentando normalmente um nível lógico alto (1). Quando ligadas ao potencial de terra, assumem um valor lógico baixo (0). O intervalo de leitura para cada entrada digital é de aproximadamente

1  $\mu$ s. Na Fig. 2, é mostrada de forma simplificada a interface da porta de jogos, ver [6].

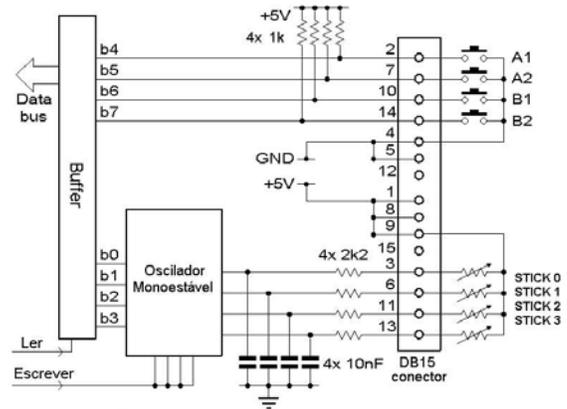


Figura 2 - Circuito típico de uma interface de joystick

O seguinte segmento de programa, em linguagem C, pode ser utilizado para verificar o funcionamento das quatro entradas digitais da porta de jogos.

```

{\#}include <stdio.h>
{\#}define PortaGame 0x201 //Endereço da porta
{\#}define MASCARA2 0x10 //Entrada Digital
{\#}define MASCARA7 0x20 //Entrada Digital
{\#}define MASCARA10 0x40 //Entrada Digital
{\#}define MASCARA14 0x80 //Entrada Digital

main()
{
  unsigned char x;
  for( ; ; ) //loop infinito
  {
    x=inportb(PortaGame); //Variavel ‘x’ recebe o
    valor da porta
    printf (“\n\n”);
    if ( x & MASCARA2) printf (“Pino 2 aberto \n”);
    if ( x & MASCARA7) printf (“Pino 7 aberto \n”);
    if (!(x & MASCARA14) printf (“Pino 14 fechado
    \n”);
    if (x & MASCARA10)
    { printf (“Pino 10 aberto \n”); }
    else
    { printf (“Pino 10 fechado \n”); }
  }
}

```

## 4. Leitura óptica

A modificação realizada no experimento consistiu em acoplar um sensor óptico de passagem com um disco perfurado, posicionados no final do trilho de ar, conforme Fig. 3 e detalhe na Fig. 4. Contando-se os pulsos gerados pela passagem de luz nos furos do disco consegue-se medir a distância percorrida pelo carrinho.



Figura 3 - Bancada experimental.



Figura 4 - Detalhe do sensor óptico de passagem com o disco perfurado.

## 5. Disco perfurado

A roldana utilizada nesta experiência é composta por uma proteção de quatro furos externos e um disco fino em seu interior com oito furos, conforme mostra a Fig. 5. Possui dupla função: a primeira é a de transferir o movimento vertical, imposto pela força do peso, em movimento horizontal do carrinho. Entretanto, a passagem do cordão sobre a roldana, força esta a girar sobre seu eixo à medida que a massa pendurada na extremidade do cordão se desloca sob ação da gravidade. A segunda função consiste em capturar a informação referente ao giro da roldana, usando um sensor óptico alinhado com os furos do disco. Contando-se as interrupções do feixe de luz poderá ser inferido o movimento temporal do carrinho.

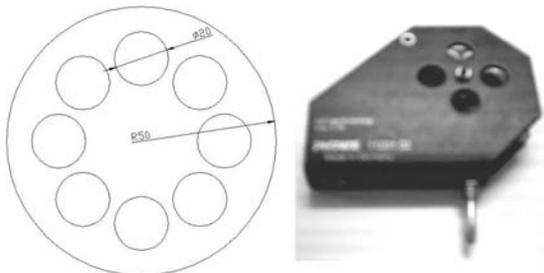


Figura 5 - Disco com oito furos: vista interna e externa.

Quando o disco perfurado roda sobre seu eixo significa que está havendo um deslocamento do carrinho no trilho da experiência. Os dados capturados pelo sensor óptico dependem da posição do disco. Assim, quando há um movimento do disco na parte onde não há furo, o feixe de luz do sensor não passa deixando o sinal lido em zero (nível baixo). Na posição em que há furo, o feixe de luz passa ficando o sinal em 1 (nível alto). No momento em que o feixe de luz do sensor atravessa o furo da roldana, o sinal alto (1) é identificado pelo *software*, mas não faz nada, porém quando o feixe é cortado e o sinal é de nível baixo (0), o computador realiza uma operação matemática para obter a posição do carrinho usando a seguinte expressão:

$$X - X_0 = \frac{(\pi \cdot D) \times (N_{pulsos} - 1)}{8} \quad (1)$$

onde  $D$  é o diâmetro externo da roldana e  $N_{Pulsos}$  é o número de pulsos emitidos pelo sensor óptico.

## 6. Programação

O algoritmo desenvolvido para a experiência da 2ª Lei de Newton foi desenvolvido usando o visual C++, ver [7]. A seguir ilustra-se o código do programa em C++.

```
{\#}include <stdio.h>
{\#}include <iostream.h>
{\#}include <dos.h>
{\#}include <conio.h>
{\#}include <stdlib.h>

{\#}define PG 0x201 //PortaGame ou Porta Joystick
{\#}define Sensor1 0x10 //Entrada Digital 1 (16)
{\#}define Sensor2 0x20 //Entrada Digital 2 (32)
//Valor capturado pela porta

unsigned char valordaporta;

float DistanciaMen=0;
float Distancia=7; //Distância percorrida
                    pelo carrinho
float pi=3.141592653589;
float D=5; //Diâmetro da Roldana (cm)
int pulso=0; //Numero de pulsos
FILE *saida; //O ponteiro coloca dados
                    no arquivo de saída

void main()
{
printf("\t DISTANCIA PERCORRIDA \n\n ");

if ((saida=fopen("Lei{\_}de{\_}Newton.xls", "w+"))
==NULL)
printf("\nERRO 'Lei{\_}de{\_}Newton.xls' \n\n");
else
{
fprintf( saida, "Distancia\n");
do
{
valordaporta = _inp(PG); //Valor da Porta
```

```

if(!(valordaporta & Sensor1)) //Primeiro Pulso -
                                START
{
    do{valordaporta = _inp(PG);
    } while(!(valordaporta & Sensor1));
pulso=pulso++;
if (pulso==1)
{
    printf("->Inicio da Captura de Dados<-\\n\\n");
}
if (pulso>=2) //Distância
{
    Distancia=((pi*D)/8)*(pulso-1);
    Distancia=Distancia; }
printf("%3dpulso,d=%8.3fmm\\n",pulso,DistanciaMen);
    //Escreve na Tela
DistanciaMen = Distancia/100;
    //Converte Milímetros para Metros
fprintf( saida, "%f\\n", Distancia);
    //Escreve no Arquivo
}
}while((valordaporta & Sensor2));
    //Termina quando o Sensor2 receber o sinal
}
fclose(saida); //Fecha o Arquivo Gerado
}

```

Para calcular o tempo entre os pulsos emitidos pelo sensor óptico utilizou-se a biblioteca “TIMER.H” que por sua vez pode informar o tempo com resolução em milissegundos. No código do programa mostrado, observa-se o uso de um segundo sensor óptico (sensor 2) para indicar o fim da experiência (fim da coleta de dados). Observar que o programa gera um arquivo com extensão .xls, o qual poderá ser aberto pelo Excel.

## 7. Resultados

A Fig. 6 ilustra o gráfico de posição *vs.* tempo, obtido utilizando o sistema projetado. São mostrados um total de 35 pontos, o qual resultou suficiente para a correta interpretação da experiência.

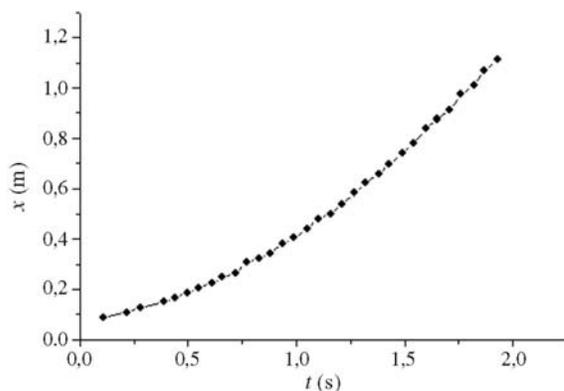


Figura 6 - Gráfico da distância *vs.* tempo obtido pelo sistema desenvolvido.

Os alunos poderão observar durante o decorrer da experiência que a distância varia com o quadrado do tempo e, desta forma, inferir sobre a velocidade e a aceleração ao linearizar a curva mostrada na Fig. 6. Ainda, para evitar que o sistema continue a medir indefinidamente e seja interrompido quando a força peso deixa de agir, devido à massa pressa na extremidade do cordão colidir com chão, testaram-se duas alternativas. Uma delas foi utilizar um segundo sensor óptico de passagem, com o qual o operador pode limitar a distância percorrida pelo carrinho, colocando-o onde ele desejar. Na segunda alternativa, o programa seria obrigado a parar após a roldana ter percorrido uma distância aproximada de 1,20 m, sendo esta a distância equivalente à altura da bancada.

## 8. Conclusão

Com o sistema projetado, a experiência tornou-se didaticamente mais eficiente, agradável e com melhor resolução, em relação à realizada manualmente pelos alunos, propiciando a oportunidade para acrescentar variações conforme a criatividade. Na solução adotada, as leituras das distâncias percorridas pelo carrinho são mostradas instantaneamente num gráfico de posição *vs.* tempo na medida em que o experimento acontece (demora em torno de 2 a 3 segundos). Cabe salientar que para aumentar a resolução na leitura da distância percorrida pelo carrinho, poderia construir-se um disco com mais furos, mas essa solução estaria condicionada ao intervalo de tempo que poderia ser medido entre dois pulsos consecutivos gerados pelo sensor óptico de passagem. Para finalizar, pode-se destacar que é possível selecionar o formato do arquivo de saída de interesse para análise, por exemplo, o Origin, Matlab, ou Excel.

## Referências

- [1] D. Halliday, R. Resnick e J. Walker, *Fundamentos de Física* (Editora de Livros Técnicos e Científicos S.A, Rio de Janeiro, 1996), v. 2.
- [2] Phywe - Manual do Fabricante, *Roteiros de Experiências* (Göttingen, 1998).
- [3] R.A. Serway, *Física* (LTC, Rio de Janeiro, 1996), v. 2.
- [4] D.F. Sousa, Rev. Bras. Ens. Fís. **20**, 413 (1998).
- [5] M.A. Cavalcante e C.R.C. Tavoraro, Rev. Bras. Ens. Fís. **22**, 421 (2000).
- [6] R. Haag, Rev. Bras. Ens. Fís. **23**, 176 (2001).
- [7] Harvey M.Deitel e Paul J. Deitel, *C++: Como Programar* (Editora Bookman, 2001).